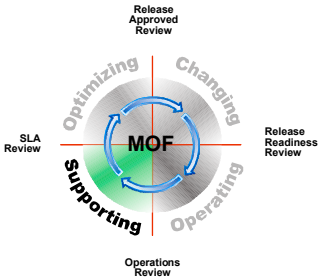
Microsoft®

MOF Service Management Function Incident Management

patterns & practices



Microsoft®

Solutions for Management

Contents

Document Purpose

This guide provides detailed information about the incident management service management function (SMF) for organizations that have deployed, or are considering deploying, Microsoft technologies in a data center or other type of enterprise computing environment. This is one of the more than 20 SMFs defined and described in Microsoft® Operations Framework (MOF). The guide assumes that the reader is familiar with the intent, background, and fundamental concepts of MOF as well as the Microsoft technologies discussed.

An overview of MOF and its companion, Microsoft Solutions Framework (MSF), is available in the *Introduction to Service Management Functions* guide. This overview guide also provides abstracts of each of the service management functions defined within MOF. Detailed information about the concepts and principles of each of the frameworks is also available in technical papers available at http://www.microsoft.com/solutions/msm/.

Executive Summary

All organizations experience incidents that either impact or threaten to impact the normal running of the business. As businesses have become increasingly dependent upon their IT services, the need to react quickly and effectively to any incidents that adversely affect IT services or infrastructure has become paramount.

Incident management is a critical process that provides organizations with the ability to first detect incidents and then to target the correct support resources in order to resolve the incidents as quickly as possible. The process also provides management with accurate information on the incidents impacting the organization, so that they can identify the required support resources and plan for their provision.

By utilizing the incident management process, organizations can ensure that their support resources are focusing on the issues that have the greatest urgency and potentially the greatest impact on the business. Without the control and management information provided by this process, organizations cannot be sure that their often-substantial investment in IT support is really meeting its objectives.

Key benefits of incident management are:

- Timely resolution of incidents, resulting in minimized business impact.
- Improved utilization of support resources.
- Better understanding of the impact of incidents on SLA targets, allowing improved prioritization.
- Accurate information on the incidents that are occurring.
- Elimination of "lost" incidents and service requests.
- Increased availability of management information.

This document illustrates the processes and activities involved in incident management, as well as the ways in which incident management relates to other solution management functions (SMFs) in the Microsoft® Operational Framework (MOF) process model.

Process and Activities

Incident Management Overview

The incident management process aims to ensure that incidents are detected and service requests are then recorded. Recording ensures that there are no lost incidents or service requests, allows the records to be tracked, and provides information to aid problem management and planning activities. The process includes the use of technology to provide self-service facilities to customers, providing them with flexible and convenient interfaces to the support function while also reducing the workload and personnel requirements of the service desk.

Service requests, such as a request for change (RFC) or a batch job request, are also recorded and then handled according to the relevant processes for that type of service request.

Incidents undergo classification to ensure that they are correctly prioritized and routed to the correct support resources. Incident management includes initial support processes that allow new incidents to be checked against known errors and problems so that any previously identified workarounds can be quickly located.

Incident management then provides a structure by which incidents can be investigated, diagnosed, resolved, and then closed. The process ensures that the incidents are owned, tracked, and monitored throughout their life cycle.

There may be occasions when major incidents occur that require a response above and beyond that provided by the normal incident process. Incident management includes a process for handling these major incidents, including management and functional escalations, effective communications, and formal rollback plans.

Goals and Objectives

The primary goal of the incident management SMF is to restore normal service operation as quickly as possible and to minimize the adverse impact on business operations, thus ensuring that the best possible levels of service quality and availability are maintained. Normal service operation is defined as a service operation within service level agreement (SLA) limits.

The objectives of incident management are:

- To restore normal service as quickly as possible.
- To minimize the impact of incidents on the business.
- To ensure that incidents and service requests are processed consistently and that none are lost.
- To direct support resources where most required.

- To provide information that allows support processes to be optimized, the number of incidents to be reduced, and management planning to be carried out.

Scope

Incident management handles all detected incidents and all service requests that can be raised through the service desk.

ITIL defines an incident as: Any event that is not part of the standard operation of a service that causes, or may cause, an interruption to, or a reduction in, the quality of service.

Typical incidents could include:

- A service being unavailable
- Software corruption
- A hardware failure
- The detection of a virus

The range of different service requests received by the IT organization varies between different organizations. Common service requests can include:

- Requests for change (RFCs)
- Requests for information (RFIs)
- Procurement requests
- Batch job requests for a specific purpose
- Service extension requests
- Password resets

Depending on the size and structure of the organization, some of these requests will be wholly processed by the service desk, while others will be handled by processes within other SMFs or other parts of the organization. In the latter case, the incident management process acts as an interface to the relevant process. An example of this is an RFC that might be passed to the change management process.

Key Definitions

The following are key terms within the incident management process:

Incident. Any event that is not part of the standard operation of a service and causes, or may cause, an interruption to, or a reduction in, the quality of service.

Initial support team. The team that provides the very first line of support for processing incidents and service requests. The initial support staff is responsible for trying to resolve incidents at first contact—by identifying known workarounds, using

diagnostic scripts, or their own knowledge. In many organizations, the service desk acts as the initial support team.

Known error. An incident or problem for which the root cause is known and a temporary workaround or a permanent alternative has been identified. If a business case exists, an RFC will be raised, but—in any event—it remains a known error unless it is permanently fixed by a change.

Major incident. An incident with a high impact, or potentially high impact, which requires a response that is above and beyond that given to normal incidents. Typically, these incidents require cross-company coordination, management escalation, the mobilization of additional resources, and increased communications.

Problem. The undiagnosed root cause of one or more incidents.

Resolver groups. Specialist teams that work to resolve incidents and service requests that initial support cannot resolve themselves. Support team structures vary between organizations, with some using a tiered structure (second, third, and so forth), while others use platform or application-oriented teams (mainframe team, desktop team, network team, or database team).

Service desk. A function that provides the vital day-to-day contact point between customers, users, IT services, and third-party organizations. The service desk not only coordinates the incident management process, but also provides an interface into many other IT processes.

Service request. Requests for new or altered service. The types of service requests vary between organizations, but common ones include requests for change (RFC), requests for information (RFI), procurement requests, and service extensions.

Solution. Also known as a permanent fix. An identified means of resolving an incident or problem that provides a resolution of the underlying cause.

Workaround. An identified means of resolving a particular incident, which allows normal service to be resumed, but does not actually resolve the underlying cause that led to the incident in the first place.

Ownership, Tracking, and Monitoring

The diagram below shows the incident life cycle from the initial occurrence through to closure of the incident following confirmation that the issue has been resolved.

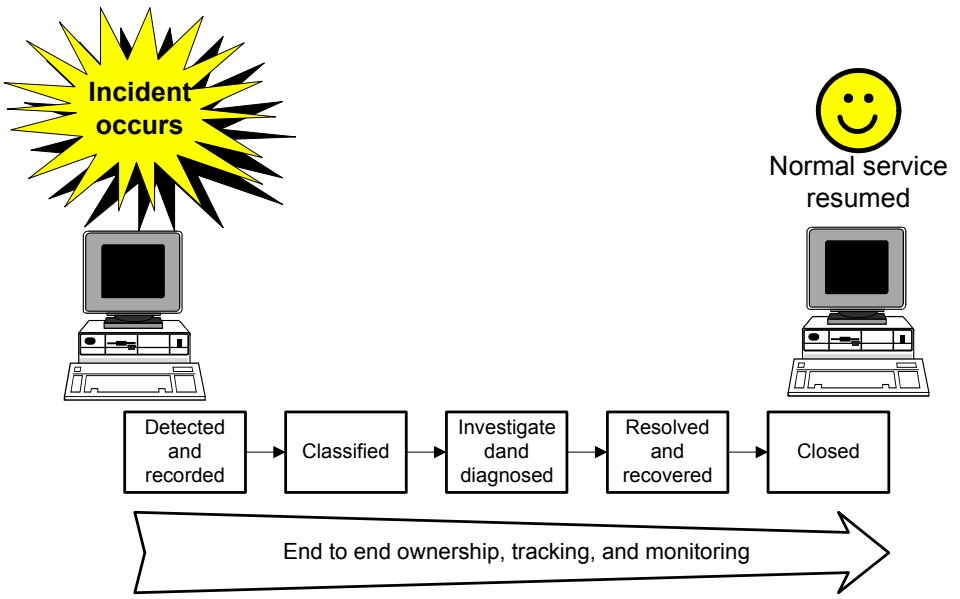


Figure 1
The incident life cycle

A fundamental concept within incident management is the end-to-end ownership, tracking, and monitoring of incidents. The service desk effectively owns all of the incidents and is responsible for monitoring their progress towards resolution. The service desk should frequently update the incident records (so that progress can be tracked) and inform the users about the progress being made.

Where it is believed that progress in resolving an incident has stalled, the service desk needs the authority to pursue and, if necessary, escalate the incident to ensure that resolution is achieved within service targets. Tools allow the service desk analysts to be automatically notified when incident records have not been updated for a certain period, which may indicate that the incident has stalled.

Major Processes

Incident management can be graphically presented in the form of a process flow diagram, which identifies the activities needing to take place in order to ensure that IT incidents are being reacted to by the correct support resources, within the timescales required by the business.

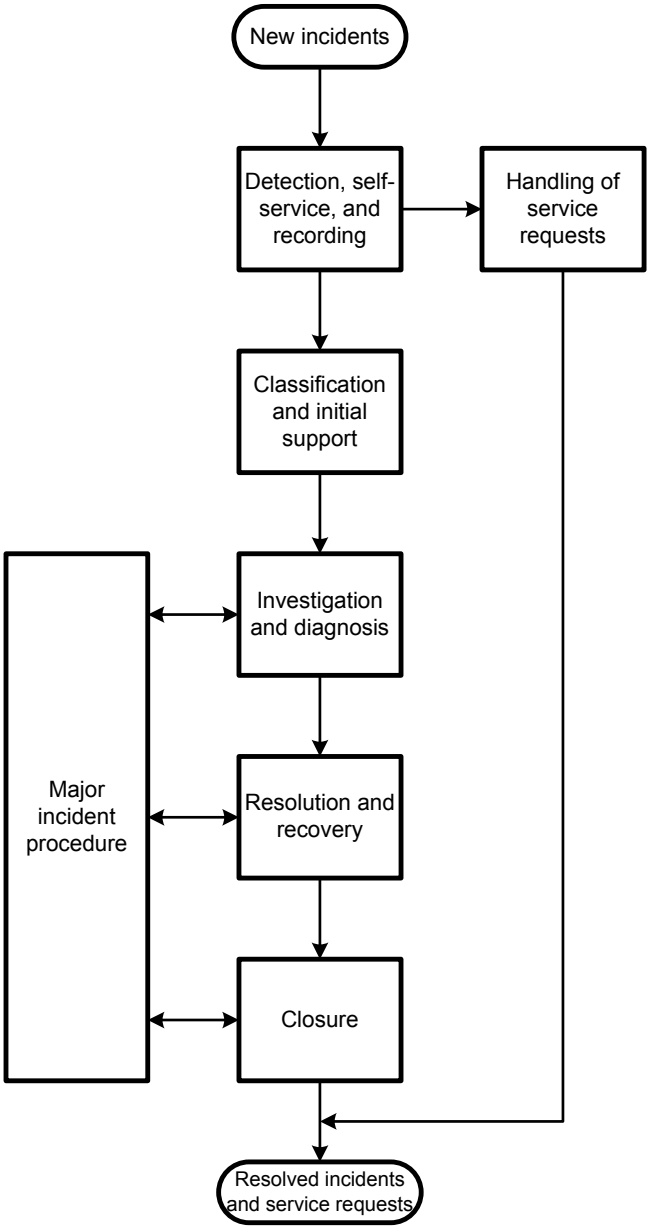


Figure 2
Incident management process flow chart

The major processes within incident management are:

Detection, Self-service, and recording. This process deals with the initial detection of incidents through a variety of different mediums. Incidents might be reported by people contacting the service desk by telephone, fax, or e-mail. Other incidents may be raised because of alerts from event management systems.

Users may utilize self-service facilities to raise their own incidents, check progress on existing incidents, and access self-help information.

All incidents are recorded, so that they can be tracked, monitored, and updated throughout their life cycle. This information can then be

utilized for problem management, reporting, process optimization, and planning purposes.

Handling of service requests. This process allows the processing and distribution of service requests. Different types of service requests require handling in different ways. The service desk may be able to process certain requests, while other requests need to be passed to other processes, such as change management, for processing.

Classification and initial support. The classification process categorizes the incident and uses information on impact and urgency to determine the priority of the incident.

The process of initial support aims to provide first-line resolution for incidents. This can be achieved by checking them against known errors, existing problems, and previous incidents in order to identify documented workarounds.

Investigation and diagnosis. This process deals with the investigation of the incident and the gathering of diagnostic data. The aim of the process is to identify how the incident can be resolved as quickly as possible.

The process allows for management escalation or functional escalation if either becomes necessary in order to meet SLA targets.

Major incident procedure. The major incident procedure exists to handle those critical incidents that require a response above and beyond that provided by the normal incident process. Although these incidents still follow the normal incident life cycle, the triggering of the major incident procedure provides the increased coordination, escalation, communication, and resources that these high-priority events require.

Resolution and recovery. This process covers the steps required to resolve the incident, often by interfacing with the change management process to implement remedial actions. Once actions have been taken, the success of the resolution is checked.

Following resolution of the incident, such as replacement of a faulty hard disk, there may be recovery actions that need to be taken, such as the restoration of data and restarting of the service.

Closure. This process ensures that the customer is satisfied that the incident has been resolved prior to closing the incident record.

The process also checks that the incident record is fully updated and assigns a closure category.

Detection, Self-Service, and Recording

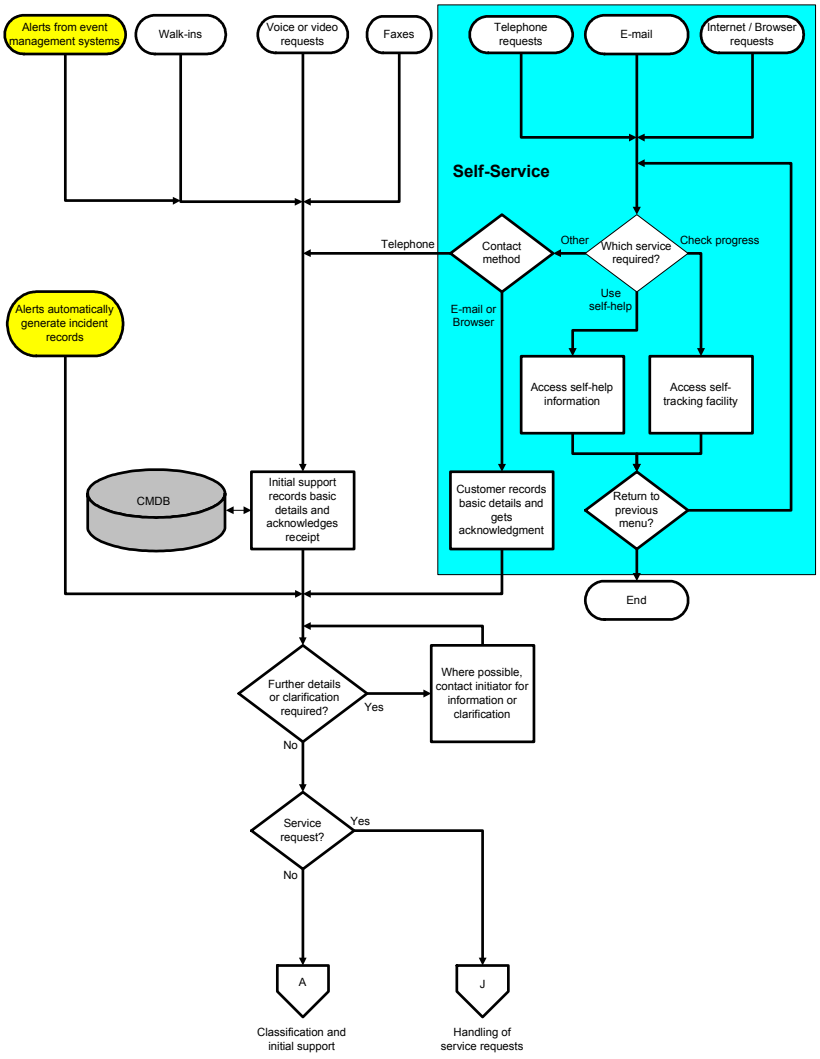


Figure 3

Detection, self-service, and recording flow chart

Detection

In order for incidents to be managed, they first have to be detected. Traditionally, the majority of incidents are reported by users who experience an issue while trying to perform their daily tasks. The service desk plays a key role in this by acting as the single point of contact between users and IT. This single point of contact helps to ensure that all reported incidents and service requests are treated consistently, while also minimizing interruptions to support staff, thereby allowing them to function more efficiently.

Incidents might also be detected by IT staff, partners, and suppliers. The ways in which such incidents can be reported may be by telephone, fax, e-mail, Internet, browser, and by those who simply walk into the service desk area.

The widespread use of event management systems now provides another source of detected incidents. These systems continually monitor systems and network infrastructures and can identify when predetermined thresholds are exceeded or components are unavailable. The systems then alert the incident management process.

Technology note: There is a wide range of event management systems in use within organizations. These include in-house scripts developed to provide basic monitoring of single systems and applications, commercial systems management products, and complex enterprise management solutions.

Some solutions are aimed at specific platforms while others offer multi-platform functionality. The selection of the correct tools is critical for an organization, as these tools often represent a significant investment in terms of both cost and setup. Many tools were originally designed for a specific platform. Although they now offer multi-platform support, the quality of that support may not equal that of the original platform.

Industry-recognized, high-quality event management systems should be used for strategic platforms. A good cross-platform management system should be able to consolidate the events from one or more products to ease the burden of harried operators. Once purchased, they can also provide monitoring for systems that may not be deemed strategic enough to warrant their own custom monitoring tool.

Careful consideration needs to be given to the thresholds being monitored. Different thresholds may be required for different purposes. A certain metric may have a threshold that corresponds to the targets set within SLAs so that SLA breaches can be identified. However, incident management requires a lower threshold of alerts so that action can be taken *before* the SLA is breached. Often, a period of tuning is required to identify thresholds that allow time for action to be taken, but which are not triggered too frequently during normal operation.

Self-Service

Overview

Self-service facilities provide IT customers with increased flexibility and control over how and when they interface with the support organization. In a self-service environment, customers are able to contact the service desk and transact business by using a variety of methods. The choices may include conventional telephone equipment, wireless devices, and conventional keyboard devices, using either browser technology or access to enquiry facilities embedded in the service desk tool. Irrespective of the method used, controls need to be in place to ensure a consistent level of service quality. There are differences that need to be recognized and planned for when dealing with different access mediums, and careful design is essential for the concept to work effectively.

To make this an effective strategy, the service desk needs to identify the types of requests that they deal with, decide which request types suit a self-service solution, and then develop the solution. If this is done well, organizations are able to free their technical resources to focus on more complex or exceptional issues.

For example, if 30 percent of the requests received are for password resets, then the development of a facility to allow users to answer

authentication questions and then reset their own passwords is worthwhile. If it is then found that a large proportion of the remaining calls are "How do I?" type questions, then development of a frequently asked questions (FAQ) information facility could be undertaken. The key is to identify request types that are currently using significant initial support resources, but which would work well as a self-service solution. This can then allow resources to concentrate on the remaining request types, such as server or network difficulties. It is necessary to understand the profile of calls being received in order for self-service solutions to be properly targeted. A matrix such as the one that follows should be constructed.

Table 1 - Self-Service Matrix

Request type	Percentage of total requests	Resolved by self-service (tier 0)	Resolved by initial support	Resolved by 2nd line resolver groups	Resolved by 3rd line resolver groups
Password resets	30%	0%	25%	5%	0%
How do I?	18%	0%	9%	8%	1%
Progress updates	17%	0%	15%	2%	0%
Office suite issues	13%	0%	5%	5%	3%
Personal computer issues	12%	0%	3%	7%	2%
Service connection issues	5%	0%	0%	2%	3%
Other	5%	0%	3%	1%	1%
Total	**100%**	**0%**	**60%**	**30%**	**10%**

From the previously shown matrix, observe that self-service solutions for password resets, FAQs, and progress updates are worthwhile investments. This allows initial support resources more time to try to solve some of the more complex incidents currently being passed to second- and third-line resolver groups. Following the implementation and marketing of the new self-service facilities, an updated matrix may show:

Table 2 - New Self-Service Facilities Matrix

Request type	Percentage of total requests	Resolved by self-service (tier 0)	Resolved by initial support	Resolved by 2nd line resolver groups	Resolved by 3rd line resolver groups
Password resets	30%	28%	2%	0%	0%
How do I?	18%	10%	7%	1%	0%
Progress updates	17%	15%	2%	0%	0%
Office suite issues	13%	0%	10%	3%	0%
Personal computer issues	12%	0%	7%	5%	0%
Service connection issues	5%	0%	1%	3%	1%
Other	5%	0%	4%	1%	0%
Total	**100%**	**53%**	**33%**	**13%**	**1%**

The matrix shows that the self-service facilities have now taken a large proportion of the requests away from the support staff. This allows each tier more time to concentrate on more challenging incidents that they would previously have escalated due to lack of resources. Note that the percentage of calls being resolved by initial support has reduced; this is because they are now dealing with request types that take longer to fulfill than simple progress updates or password resets. Having time to work on more challenging incidents increases job satisfaction for the staff, while also reducing the day-to-day impact on higher tier resources. This allows the staff to concentrate on the more strategic issues.

Some types of requests do not suit a self-service approach: Due to its nature, either the request requires a response from specialist support personnel or it needs to be tracked and its metrics collected.

High-priority incidents should not normally be reported through a self-service mechanism, but instead should be reported directly to the service desk analysts. This helps ensure that all high-priority incidents are correctly classified and that all necessary details have been gathered from the initiator.

As it is more difficult to collect metrics on requests being resolved through the use of self-help facilities, in most circumstances the contents of these should be aimed at answering “How do I?” type requests rather than resolving failures.

Contact Methods

For the self-service solution to be successful, it must be easy to use so users feel confident about using it and believe that it offers them a valuable alternative to contacting the service desk by other means.

Regardless of how the customer enters the self-service environment, their entry needs to be validated and metrics need to be collected. This includes trend analysis of the enquiries made (both successful and unsuccessful), the volumes of contacts made, the performance of the self-service environment, and changes in the method of contact. This information helps identify improvements in the service, but it also demonstrates the effectiveness of the service and provides information on the return on investment.

Key data to collect includes:

- Means or mode of entry
- Date and time of entry
- User identification
- Questions asked and facilities accessed

Depending on the environment, various authorization checks may need to be made to ensure that the caller is authorized to access the facilities and data. This is especially pertinent for outsourcing organizations supporting a number of customers, who should not be able to view each other's support requests. Authorization checks may involve correctly answering one or more security questions (such as passwords or mother's maiden name). Procedures need to be established to collect, securely store, and then maintain this security validation information.

Technology note: Security information to be used during validation checks needs to be stored securely in a database. This information may either be linked to or form part of the user record within the service desk tool. Initial checks might involve verifying user specific information held in the user record, such as the user's ID number, home address, or e-mail address.

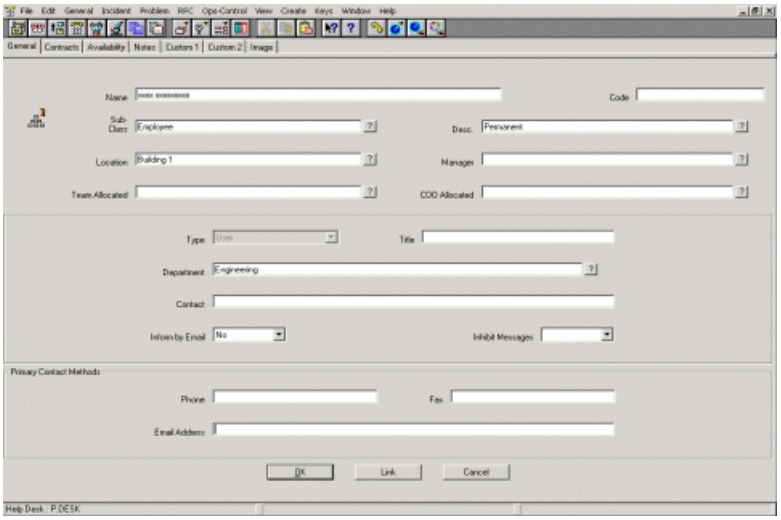


Figure 3
A record holding user details

Type of Facilities

The mode of entry determines the facilities available to the inquirer. If computer-based, such as a keyboard, then standard Web interface and search engines should be provided. These may be integrated with the service desk tool or based on a third-party inquiry tool.

If the mode of entry is a telephone-style keypad, then more specific and scripted interaction needs to be deployed and voice recognition technology may be required.

Other modes of entry need to be considered and catered for, such as e-mail, where predefined task forms may be appropriate for logging new service requests or requesting progress updates. This method of communication is likely to be less successful for provision of dynamic help.

A typical high volume, but relatively straightforward, request might concern password resetting. With a scripted self-service solution, service desk analysts no longer need to be involved in resetting passwords. Employees who need their passwords reset can access a self-service facility, answer several authentication questions, and proceed through a guided process to reset their own password. It is always advisable when performing password changes to send an advisory e-mail to the caller as a security precaution.

New requests made through the self-service facilities need to be captured in the service desk system and, as with all other requests, be

allocated an incident number to provide an audit trail and enable subsequent progress monitoring. This linkage with the service desk is particularly important when self-service allows callers to change data or passwords. It then ensures that self-service calls are automatically included in trend analysis. Specific analysis of this data allows the service to be improved and the trend analysis to be comprehensive.

The information available depends upon the style of inquiry. If the inquiry is computer- or Web-based, then access to a variety of data sources is possible, including:

- Incident and problem tracking
- Forward schedule of changes
- Supplier support notes
- Training material
- Software updates
- FAQ information
- Service catalogs
- Price lists

The content, accuracy, and timeliness of the information need to be managed carefully. Clear roles and responsibilities need to be defined and agreed upon to ensure no ambiguity. Information available through any form of self-service facility needs to be closely monitored and audited to ensure compliance with regional regulations—with particular attention being paid to data protection and confidentiality agreements.

Technology note: Using browser technology to support self-service may increase traffic on the organization's network, so it is important to monitor the capacity of the network and its components. Voice recognition technology is improving continually and is increasingly likely to be used in self-service facilities.

Accessing Self-help Information

If the user elects to use the self-help facility, he or she should be given every opportunity to find an answer quickly and with as little confusion as possible. If users find the experience distasteful, it is unlikely that they will use it in the future. Just like any other product, the users need to "like" it before they will "buy" it.

Self-help systems are typically a database of useful information and experience. The user provides some information to the system in hopes of finding a "match" to their issue. This information can be an error code, an error message, or a freeform description of the issue.

If a match is found, the information is normally returned in the same format as the original inquiry. It is possible, however, that the user may require a printed response. For example, a user may wish to

have a copy of a FAQ, so response options may need to be provided that include sending an automated e-mail to the caller.

Once the requested information is returned, the user needs to have the opportunity to refine the request based on the current inquiry, to select additional menu options, or to start a new request. When designing scripts, it is vital to perform full testing to ensure that a user is never left without a valid or appropriate choice; including the ability to return to the non-self-help option at any point.

If a user wishes to make a second request, the process should provide the user with the opportunity to enter another query without having to register again. Options available here need to be evaluated carefully as there may be security issues to consider. While automated systems generally may be more controlled in that security validations have to be precisely met, it might be preferable to handle requests for increased access to applications through personal contact.

If users are unable to find what they are looking for or have reached the end of the available options, they should be asked if they wish to contact the service desk. Contact with the service desk can range from simply leaving a message to a full request for the desired information. Consideration needs to be given to the methods by which this type of communication is monitored and managed, but it is imperative to record that this request has already been through a failed self-service request, so that improvements to the self-service environment can be made, if possible.

Close working links with incident and problem management are clearly required here, especially if the focus of the self-service facility is to reduce the number of simple, common support calls, such as changing a printer toner cartridge. The primary focus should be on providing an efficient, customer-oriented inquiry facility that enables commonly asked questions to be handled without the involvement of the service desk. The information must be constantly checked to ensure it is accurate and reflects the current state of support (and service) information.

Scalability

A self-service solution allows a very flexible response in organizations when rapid growth is experienced. Typically, it can take several weeks to recruit and train service desk staff in response to increased demand, such as those arising from mergers. Self-service technology, however, is able to fulfill much of the demand for increased service without such time constraints, but it is essential that the system being used is capable of scaling to meet future demand.

Staffed telephone-based services are the most expensive method of providing support because of the personnel required. The technology and equipment is also expensive to install and maintain, but it is

likely that there will always be a need for staffed, telephone-based services in order to deal with the exceptions, as opposed to routine support requests. Making automated responses the primary response method requires significant initial investment but also provides a rapid and realizable return on investment. Current assessments suggest that well-designed, Web-based self-service can cost less than 5 percent of equivalent conventional support.

Most scientific studies show that people prefer a limited interaction (meaning a short duration) with a non-human interface, and most people prefer human contact. However, properly targeted, designed, and marketed self-service solutions can be seen as advantageous by customers—as has been the case with bank automated teller machines (ATMs). The key reasons for the success of ATMs is they are convenient and easy to use, extend the normal service time by being available 24/7, and often provide a quicker alternative to standing in line to see a human teller. Organizations introducing self-service should learn from this example and aim to provide solutions that are also convenient and easy to use, offer extended services, and are at least as quick as waiting for a human contact.

As an organization moves toward self-service, however, a requirement for sophisticated maintenance procedures emerges. If self-service is to be adopted, it must be done well from the beginning.

Recording

All detected incidents and service requests need to be recorded so that they can be tracked to ensure that none is lost. Recording all contacts also provides the information needed to initiate actions and reduce specific types of calls. For example, a service desk might find that a large percentage of the incoming calls are from people simply requesting telephone numbers. An investigation might reveal that the internal telephone directory booklets are significantly out of date. A decision can then be made to publish the internal directory on the intranet where it can be more easily maintained. This initiative not only reduces call numbers to the service desk, but also increases business efficiency by allowing staff easy access to up-to-date information. If the telephone number requests had not been recorded, then the scale of the problem would have been difficult to ascertain and might have gone on indefinitely without being addressed.

The incident details should be recorded in a database within the service desk tool. For each incident, an incident record should be populated with the following information:

- Unique reference number
- Date and time recorded
- Identity of the person recording the incident

- Identity of the person reporting the incident (including name, department, location, and contact details)
- Contact method
- Details of affected configuration items (CIs)
- Description of the symptoms and any error codes
- Steps needed to reproduce the difficulty

For service requests, much of the preceding basic information is still required, although the description of the symptoms should be replaced with details of the service being requested. Some types of service requests may have their own tool for recording the request, such as a change management system for RFCs, while other requests, such as unique batch job requests, are less likely to have specific repositories and should, therefore, be recorded as incident records for tracking purposes.

The call initiator identity information (such as name, department, location and contact details) should be checked against records held in a customer database, preferably forming part of the configuration management database (CMDB). This procedure allows the records containing customer details to be kept up-to-date. Depending on the environment, the procedure can also include asking questions about commercial details (for example, a contract number) and security verification to confirm identity.

Each incident or service request should be given a unique reference ID. Call initiators should be provided with this ID, which can be used to easily locate the correct record if the initiator calls again to update the record or check progress. Where incidents have been raised because of an event or alert from an event management or monitoring tool, the event or alert reference number should be included within the incident record. This allows personnel investigating the incident to identify and view the original event or alert.

The service desk is responsible for initially obtaining all the required information from the call initiator. There may be instances when some clarification or additional information may also be required. In these instances, the service desk should contact the call initiator to obtain the information.

Throughout the incident life cycle, an incident record may pass through a number of different states before finally being closed. A status field on the incident record is used to quickly identify an incident’s current state.

Examples of status categories are:

New. The incident has been recorded but may still require clarification.

Accepted. The incident has been fully recorded and is now with initial support.

Assigned. The incident has been assigned to a resolver group and is now waiting to be progressed.

Active or work in progress. Action is underway on resolving the incident.

Awaiting evidence. Action has temporarily halted while waiting for additional evidence to be obtained.

Scheduled. Further actions cannot be carried out until a scheduled time.

Suspended or on hold. Action has temporarily halted pending an event or time.

Resolved. The incident is believed to be resolved. The service desk needs to confirm resolution with the call initiator prior to closing the incident. If the call initiator is not satisfied with the resolution, then the status is reset to assigned or active.

Closed. The call initiator has verified that the incident is resolved and so the incident has been closed.

It is important that the incident record be kept up-to-date throughout the incident life cycle, so that all support staff can see what is currently happening, who is currently working on the incident, and what has previously been tried and discovered. The up-to-date record also allows any contacted staff to provide the call initiator with a progress update. Keeping the customer updated with progress is an important factor that directly impacts customer satisfaction. Updates should be provided on a regular basis, depending on the priority of the incident. For example, high-priority incidents may require hourly updates, while medium-priority incidents receive daily updates, and low-priority, long running incidents require weekly or even bi-weekly updates.

Following the recording process, incidents progress through the classification and initial support process, while service requests are passed to the "handling of service requests" process.

Technology note: Incident records should be recorded and tracked within a service desk tool, preferably forming part of an integrated service management tool set. The tool should provide minimal incident records for use when raising new incidents. To aid efficiency, these tools should make use of a customer database to automatically populate such fields as location, department, and contact details as the call initiator's name is being entered. Tools may also use Computer-Telephony Integration (CTI) to automatically populate fields based on the incoming telephone number. Even where these technologies are in use, details such as

contact number and location should still be checked with the customer during incident recording. This provides a routine audit of the data held within the CMDB and gives an indication of areas that may require further investigation or checking.

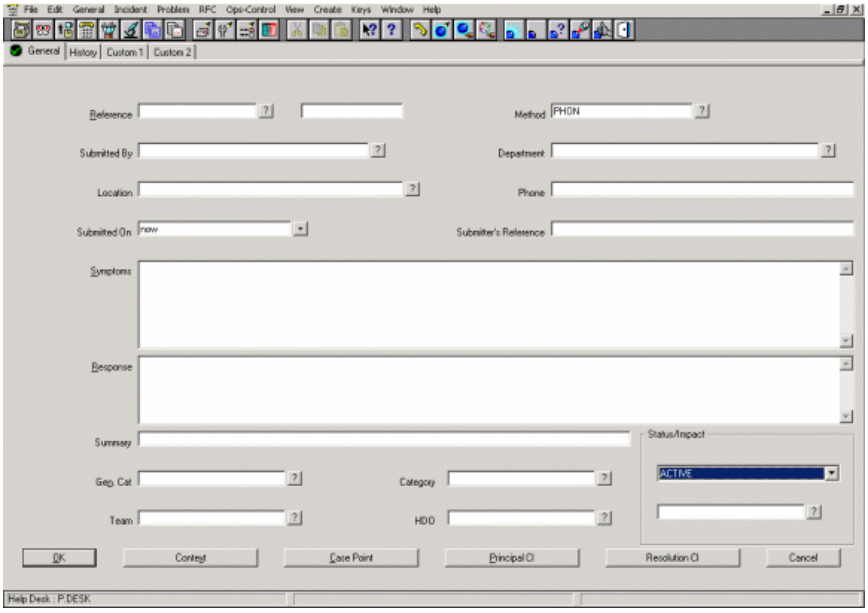


Figure 4
A typical incident record template

The service desk tool should use mandatory fields or scripts to ensure that people raising a new incident record capture all of the required basic details.

Service desk tools may allow integration with event management systems so that incidents can be automatically raised when alerts are generated. This automation takes the task of manually filling in incident records away from support staff, providing increased efficiency. Incident records created either manually or automatically following an alert need to contain sufficient information to allow the support staff to identify the events that generated the alert. If the events or alerts have unique reference IDs, these should be included within the incident record, along with the details of the configuration items (CIs) affected.

The following incident record example has been generated automatically and the submitter's reference field has been used to record the alert reference ID.

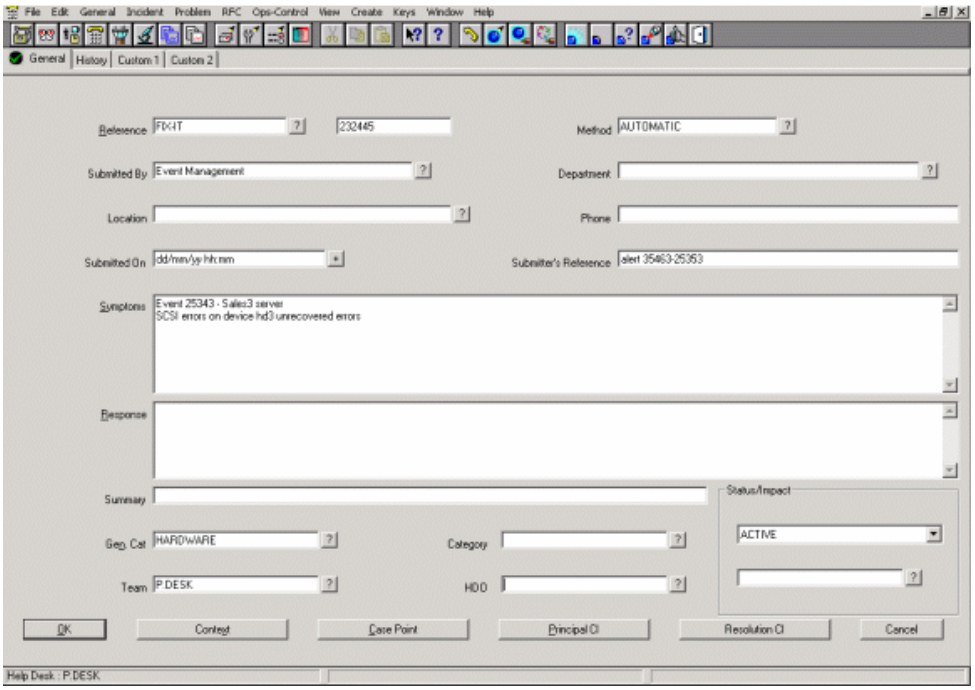


Figure 5

An incident generated by an alert

During the lifetime of an incident, the service desk tool may automatically alert the service desk analysts when customer updates are due, based on the incident's priority and the organization's policy on updates. This alert could take a number of forms:

- A clock or exclamation point icon next to the incident when the incident queue is viewed.
- The incident appearing as a certain color (such as red) when the queue is viewed.
- A pop-up dialog box.
- A message or e-mail to a designated alias.
- The output of a regularly run report (two to three times daily).

Handling Service Requests

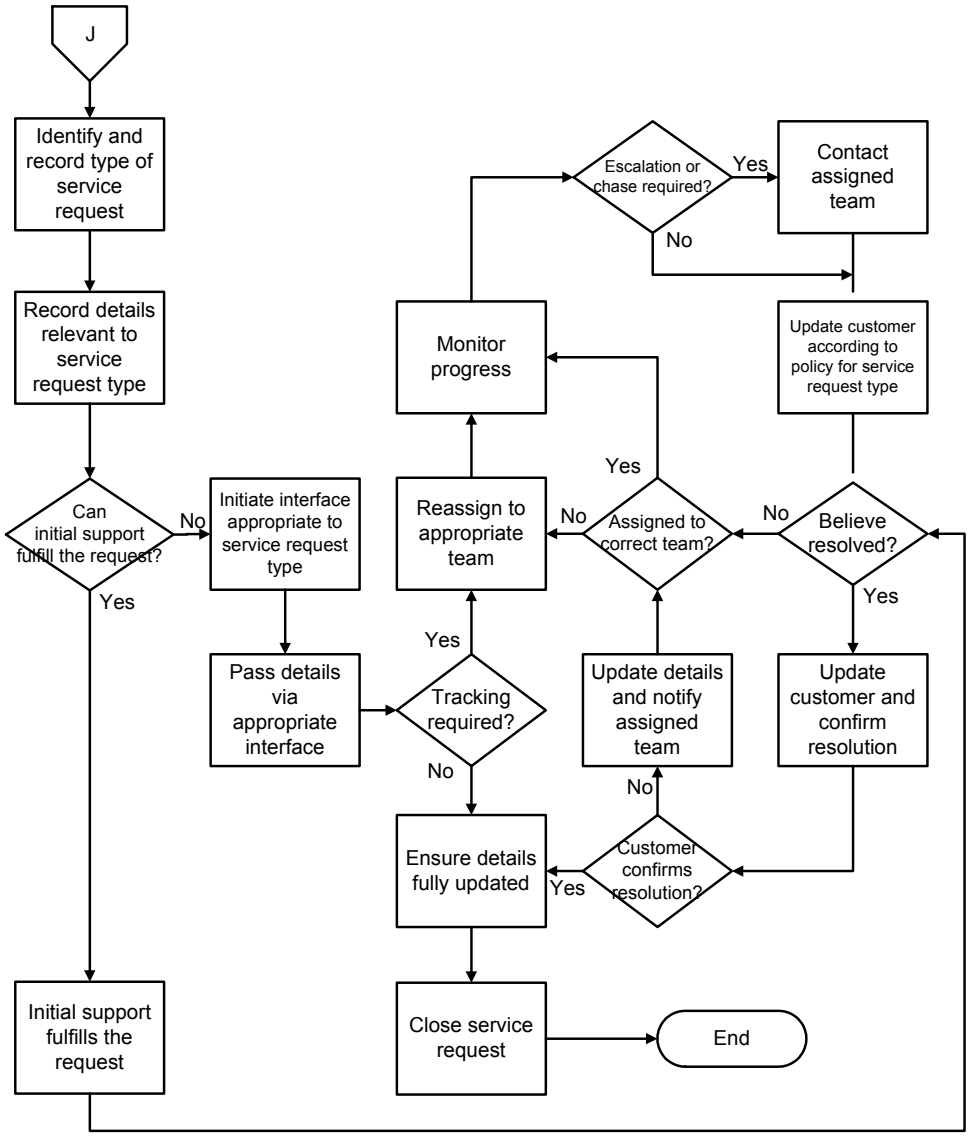


Figure 6

Handling service requests flow chart

When handling a service request, the first action is to identify and record the type of service request being processed. As mentioned previously, some types of service requests may have their own tool for recording the request, such as a change management system for RFCs. Other requests, such as unique batch job requests, are less likely to have specific repositories and should be recorded as incident records for tracking purposes.

Depending on the type of service request, different information is required from the initiator. For example, the information required in order to schedule a unique batch job will be very different from the information required for a procurement request. The service desk

needs to ensure that they have recorded all of the information required in order to process the required service request.

The type of service requests being received varies considerably between different organizations. There should be a procedure or workflow for handling each different service request type within the organization. This ensures that all requests of a particular type are handled consistently. If requests of a new type start to appear, then a corresponding procedure should be written, agreed upon, and distributed.

Many of these procedures might be simple interfaces describing how to get the new request fed into a process being performed by another SMF or part of the business. Examples of this type may be RFCs and procurement requests. The service request procedures need to ensure that all required details are recorded and then passed to the relevant process.

Other service requests may require much more complex procedures or workflows. An example is a request to set up the system for a new employee. The process might involve setting up new network and e-mail accounts, getting the employee to sign security policies, granting access to applications, arranging IT and security induction training, updating the internal telephone directory, updating the CMDB, and procuring and configuring new equipment such as a portable computer and telephone.

Technology note: For these more complex requests, it is beneficial to use a workflow management system to break the request down and track individual work actions, some of which may be occurring in parallel, while others have to wait, as they are dependent on other actions. Workflow management tools can also help with the design of the workflow procedures by creating flow charts that describe actions and dependencies, as well as the team required to perform each action and dependency.

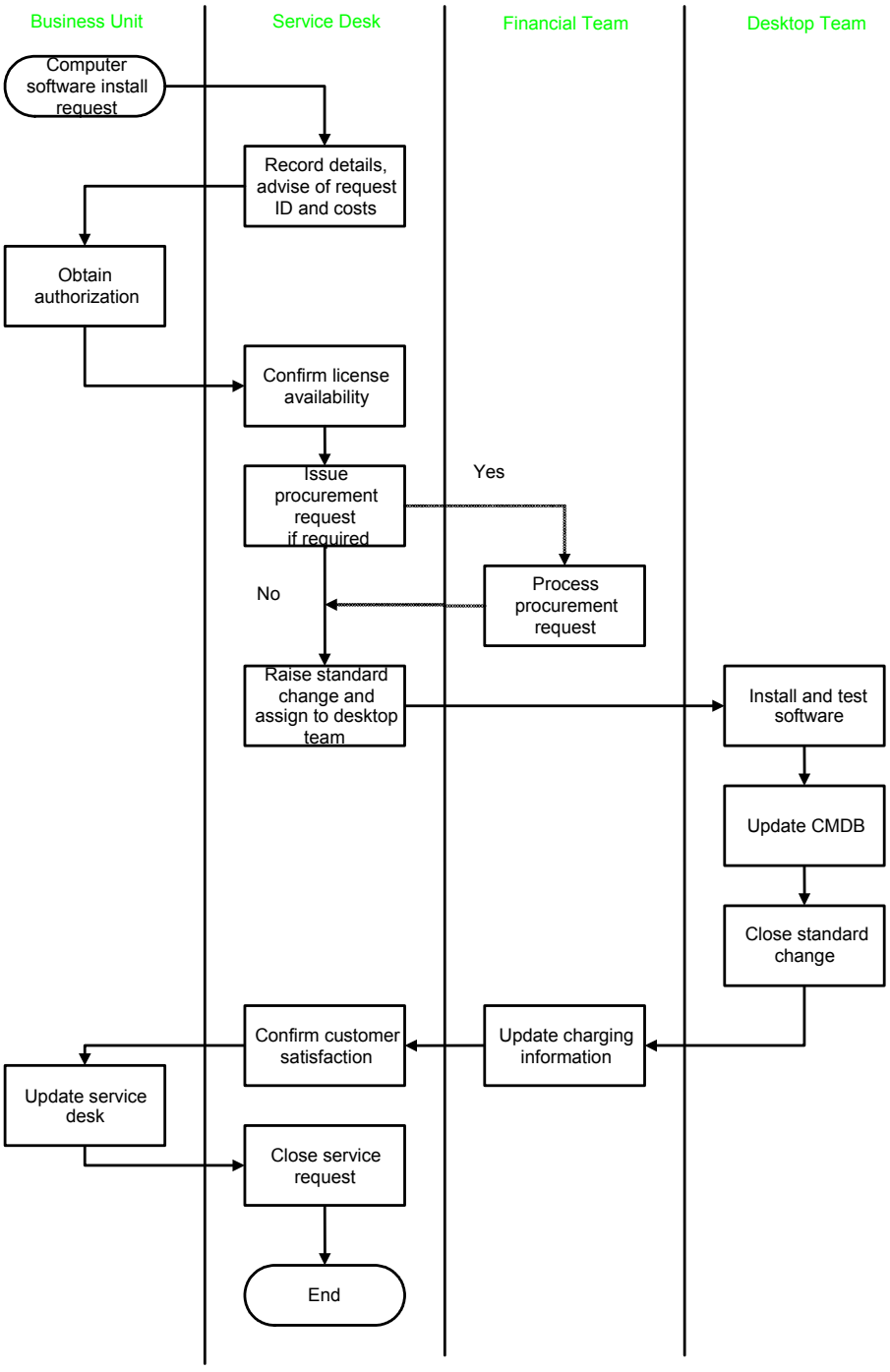


Figure 6

Sample workflow procedure flow chart

Some service requests can be wholly processed by the initial support team (often the service desk) without having to pass the request to another process or team. An example of this type of request is a request to register someone in a user-training course being run by the IT department. The procedure may allow the service desk to check availability and make the registration, without having to contact the training department.

In cases where initial support wholly processes the service request, they should perform the necessary actions, inform the initiator that the request has been carried out, confirm that the initiator is happy with the resolution, and then close the request, making certain that the record has been fully updated.

In cases where the request has to be passed to another process or team, initial support is responsible for obtaining the relevant information needed to process the request for this particular type of service and passing it on to the other process or team.

In the vast majority of cases (even though the service desk has passed on the request), the service desk still acts as the contact point if the request initiator wants to query progress or amend the request details. In these cases, the service desk should assign the request to the relevant team and then track and monitor the progress of the service request, acting as the interface between the initiator and the other process. If progress on the request seems to have slowed, the service desk confirms the current situation by contacting the team involved and, if necessary, escalating the incident to ensure that service targets are met. If at any time the request needs to be reassigned to a different team, the service desk will again carry out this action.

Once the request is thought to be resolved, the service desk should contact the initiator and confirm that he or she is satisfied with the resolution, and then close the request, making certain that the record has been fully updated. For some requests, it may be appropriate to e-mail the initiator with information that the request has been resolved, along with providing a contact at the service desk if the resolution is not satisfactory. The e-mail should advise the initiator that the resolution will be assumed resolved and the request closed if the service desk does not receive feedback within a given period (such as three weeks). Depending on the service desk tool, it may be possible to automate a standard e-mail to the initiator upon the request entering resolved status and to automatically close the request if no update is entered within the given time frame.

In cases where no service desk tracking is required (normally because the request has been immediately completed and the initiator informed), the service desk will ensure that the request record is updated and then close the request.

Technology note: Some service request types have specific tools in which they are recorded and tracked. These tools may be in-house developed solutions or third-party products, either separate from or integrated with the service desk tool. The following picture is an example of such a tool used for recording and tracking change requests.

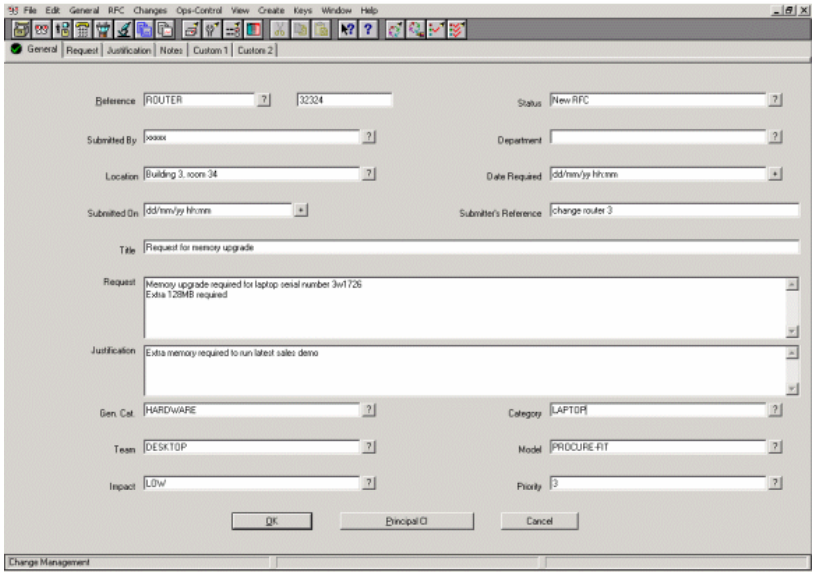


Figure 7

Example of a specific tool for creating a request for change (RFC)

Service request types for which there is no specific tool should be recorded and tracked using incident records. The following picture is an example of a batch job request in this format.

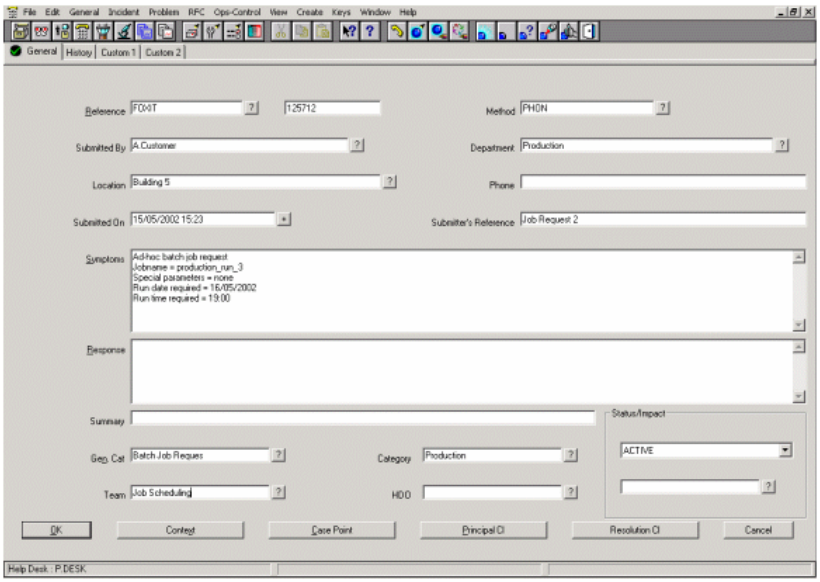


Figure 8

A unique batch job request recorded on an incident record

Classification and Initial Support

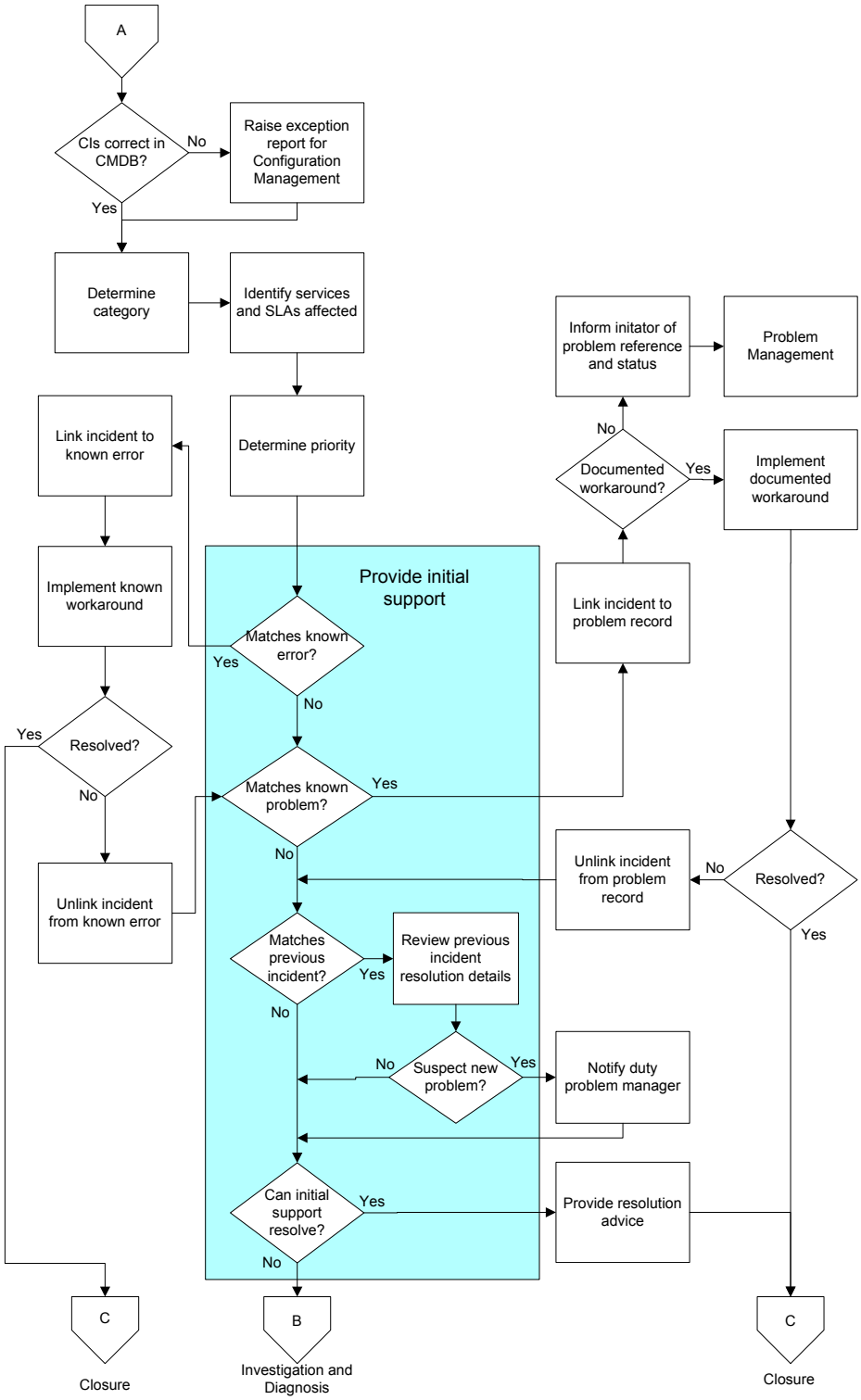


Figure 9

Classification and initial support flow chart

Classification

Overview

Incidents must be classified so they can be handled as effectively as possible with the appropriate resolution taken. Classification is the process of categorizing and prioritizing a given incident and is a very important first stage in incident management as it determines the subsequent action to be taken. Classification is used to:

- Specify the service or equipment to which the incident relates.
- Associate any relevant service level agreement (SLA), operating level agreement (OLA), or underpinning contract (UC).
- Identify the appropriate resolver group.
- Define the priority of the incident.
- Identify the workload estimate.
- Act as a matching criterion for identifying previous incidents, known errors, or known problems.

The majority of incidents may be regularly experienced—for example, password resets—so their classification will be well known. Others will require investigation before they can be classified and a number of techniques can be used to achieve this.

- The use of diagnostic scripts by service desk analysts which:
  - Controls the completeness of information gathered.
  - Uses logic and guides questioning along controlled lines.
  - Controls the terminology used to conform to support group expectations.
- Reference to CMDB records:
  - Confirms the technical detail of the customer's configuration.
  - May identify previous incidents relating to equipment.
  - Identifies use of unauthorized software.
  - Identifies any upgrades required.

Checking Against the CMDB

If there are differences identified while checking against the CMDB, it is important to raise an exception report to configuration management for corrective action to be taken. The fact of an exception might indicate unauthorized change or a breakdown in the change process. A simple exception report might be as follows:

```
Date:              April 1, 2002
CI record number:  Software1000
Recorded data:     Excel 2000 version 8.5
Reported Data:     Excel 2000 version 7
```

***Technology note*:** It is important that the service desk have direct access to the CMDB, ideally through use of an integrated service desk tool set. Integration allows the automatic population of key incident record fields from the CMDB when only the customer's name or phone number has been recorded on the incident record. Apart from enabling the service desk to take control and be proactive from a very early stage in the incident (this can enhance the service desk effectiveness and customer confidence), it prompts validation of the CMDB data and identification of any exceptions.

The search in the following picture is for a catalog (CI) number, but CIs can also be selected by name, owner, submitter, department, or location.

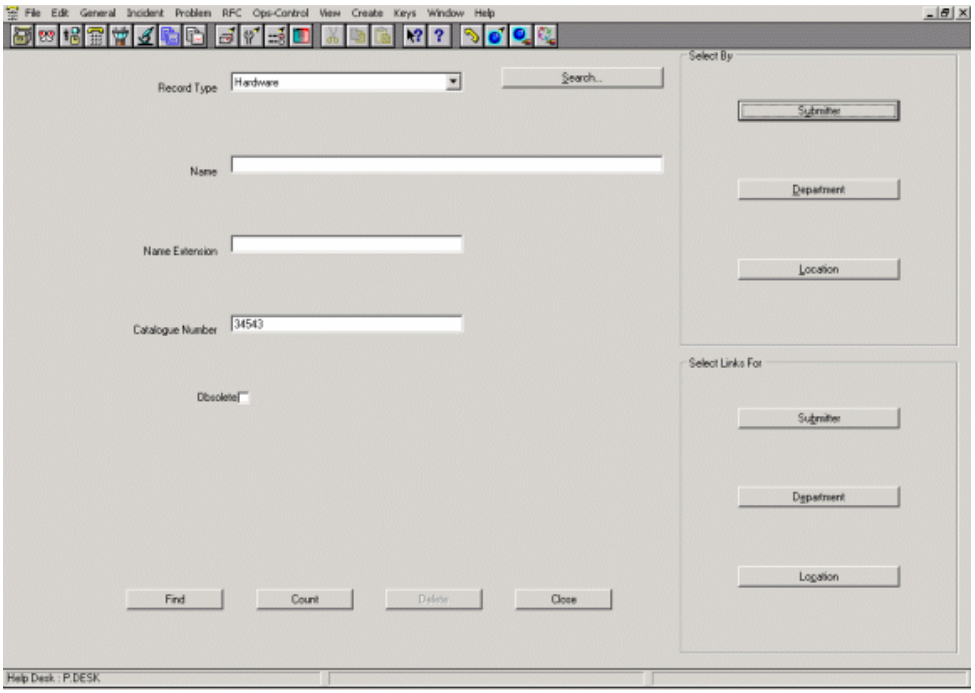


Figure 10

Example of a CMDB search screen

The benefits of integration with the CMDB are that it optimizes the speed of data access, ensures consistent use of terminology and approach, and, perhaps most importantly, ensures a single, trusted source of information. Use of multiple or un-integrated tools introduce a degree of risk and inefficiency into a key service area.

Diagnostic Scripts

Diagnostic, or troubleshooting, scripts provide assistance to the call taker when resolving incidents and typically use a structured question-response-action flow. The scripts can be set up to process common types of incidents or, if necessary, to perform front-line screening and information gathering prior to recording the call and referring it to a second-line group.

The scripts can also be used to prompt the operator to capture additional fault-specific information. For example, "Can't print" can automatically prompt for the print queue and server names, which

are then recorded as part of the call details. An example "Cannot print" script is:

1. User logs call with service desk. Symptoms may be reported as "Can't print," "Printer error messages," or "Poor print quality."
2. Confirm the user's:
   a. Name
   b. Location
   c. Telephone extension number
3. Request the printer's:
   a. Name
   b. Location
   c. Asset tag number
   d. Type (network or local)
   e. Number of people affected by the fault
4. Check CMDB record to confirm reported details and raise exception report, if required.
5. Assign a category of "printing issue."
6. Select a subcategory indicating the type of printing being performed (network or local).
7. Identify the impact, urgency, SLAs, and the services affected. Determine a priority based on this information.
8. Match against known errors, existing problems, and previous similar incidents.
9. If no related records, then a series of questions should be asked with suggested actions depending on the responses to the questions:
   f. Is the customer able to print to another printer?
   g. Is the customer able to print from another application?
   h. Is anyone else able to print to the printer?
10. If no one can print to the printer, then:
11. Ask for any error messages.
    a. Check the printer is switched on.
    b. Check that there are power and network connections.
    c. Check for any error codes on the actual printer.
    d. Try switching the printer off and then on again.
    e. Check print queue status and try stopping and starting the print scheduler.
    f. If the incident cannot be resolved, assign the call to the team responsible for print and output.
12. If the customer can print from other applications to the printer, then:
    a. Ask for any error messages.
    b. Try closing and restarting the application.
    c. Check which printer is specified as the default for that application.
    d. Re-select the print device within the application.

e. If the incident cannot be resolved, assign the call to the team responsible for the application.

13. If the customer cannot print to any printer but others can, then:
    a. Ask for any error messages.
    b. Check which printers are set up for the user.
    c. Check that the personal computer is connected to the network.
    d. Check the print queue status.
    e. If the incident cannot be resolved, assign the call to the team responsible for desktop support.

Diagnostic scripts are not just passive. They can also be used as "expert tools" where the response to a question triggers an action so that procedures and other documents are displayed automatically in the diagnosis flow. For example, if the script suggests that the user's configuration settings are incorrect, then a diagram of the relevant settings can be displayed from the CMDB. Similarly for a "How do I do this?" type of inquiry, the relevant documentation can be displayed from the online manual.

A script can run another Microsoft Windows® operating system application, for example, a "network errors" script can automatically start the network diagnostic tool set.

The uses for these active scripts are virtually limitless: They facilitate call resolution by non-expert staff as well as minimizing the number of callbacks for more information.

Categories and Priorities

The classification categories used vary from organization to organization, but they should always be agreed between the IT organization and the business community. There may be conflict between these groups as the former is primarily technical in character, while the business community may not be.

If classification categories are too technically based, then the service desk analysts must be able to translate symptoms described by business users into technical equivalents. There is more likelihood of error and difficulties in matching under these circumstances, and there may be difficulties in moving toward a self-service environment if the business community cannot easily interpret the information.

Careful selection of categories will assist with accurate referral to resolver groups if initial support cannot resolve the incident, and consistency of incident classification makes it easier to target self-service facilities.

Sample categories are shown below:

- Hardware
  - CPU

- Memory
- Hard Disk
- DVD ROM
- Monitor
- Keyboard

- Software
  - Spreadsheet
  - Word processor
  - Office application

Regardless of the categories chosen, the most important aspect is that both support and business user representatives should be involved in the initial design stage since categories become harder to change as data volumes associated with them increase.

Technology Note: When selecting a service desk tool, the design of the classification categories must be taken into account, as some tools are less flexible than others are in this regard. If an inadequate analysis tool is used, key information may be relegated to freeform text fields, which impedes trend analysis and self-service development. This is particularly important if there is no CMDB in place, as all of the analysis will be recorded in the incident record.

The service desk should determine which services and SLAs are affected by the reported incident, as this will allow an appropriate priority to be set. The CMDB should be used as a source of information when determining affected services and SLAs.

Prioritization is an important aspect of managing an incident in that it summarizes the characteristics of the incident and determines the overall approach to resolution.

An incident's priority is normally determined by the relationship between its impact and urgency, which in turn gives rise to a target resolution time. For example, a high impact, high urgency incident will receive a high priority, while a high impact, low urgency incident will receive a lower priority, as might a low impact, high urgency incident. Incorrect classification of the priority inhibits the ability of incident management to perform effectively.

The incident priority should be based on:

The services and SLAs impacted or potentially impacted. Service level agreements formalize the relationship between the service provider and the customer base. Incidents must be linked with the correct SLA to allow correct classification. This quickly and accurately establishes the incident's urgency and its impact on the business, as the SLA identifies the service affected, the users of that service, and the consequences of service interruption. It also records the expected timescales for return to normal and

the arrangements for the invocation of business continuity planning. Any associated OLAs and UCs should also be considered.

Any relative priority associated with the assigned category. Categories may be used to determine relative priorities to assist staff with determining the priority that should be given to each incident. Relative priorities should act as a starting point when determining the actual priority to be given to the incident.

Table 3 - Incident Categories and Priorities

Main category	Sub-category	Relative priority
Software	Spreadsheet	1
Software	Word processing	1
Software	Office application	2

However, this view does not distinguish between business areas, so all spreadsheet failures would be allocated the same priority, perhaps wrongly. The following table shows an alternative view that matches the classification more closely to business areas.

Table 4 - Business Areas and Priorities

Main category	Sub-category	Relative priority
Accounting	Spreadsheet	1
	Word processing	1
	Office application	2
Production	Spreadsheet	3
	Word processing	2
	Office application	2
Distribution	Spreadsheet	3
	Word processing	3
	Office application	3

The business impact and criticality. The impact of an incident is normally determined by the affect it has on the business of the organization, often known as "Business Impact." The determination of impact in this way can only be effectively undertaken in agreement with the business community during the development of the SLA. Factors that influence criticality include:

- The number of users affected.
- The extent to which business degradation results.
- The stage in the business cycle when the incident occurs.
- The impact information captured during the development of SLAs can be used to associate indicative priorities for specific categories of incidents. These priority codes should act as a basis for determining the priority, but should be modified as necessary, depending on the specific impact of the individual incident.

The urgency of the request. This relates to the speed with which an incident needs to be solved according to its impact. It is likely that a high-impact incident will also be urgent, but this is not always the case. An incident may have a high impact on an organization, but low urgency if the organization does not require the application to be available for 6 months. Alternatively, an incident could have a high urgency but low impact, such as difficulties with the personal computer used by the secretary of one of the directors. Equally, urgency may change over time: for example, in 1996 the Year 2000 problem was very high impact but not high urgency, in 1997 it was still high impact and was becoming more urgent, in 1998 the urgency increased again, and so on.

The table below shows an example priority coding system based on impact and urgency.

Table 5 - Impact versus Urgency Priority Coding System

		Impact		
		High	Medium	Low
Urgency	High	1	2	3
	Medium	2	3	4
	Low	3	4	5

Target Resolution Times

The target resolution time of an incident is the time by which the IT organization needs to have resolved the incident in order to remain within the agreed service levels. Different priorities of incidents will have different target resolution times.

The priority assigned to an incident can be used to determine target resolution times, as shown in the table below:

Table 6 – Example Priority Codes and Target Resolution Time

Priority code	Description	Target resolution time
0	Major incident	---
1	Critical	1 hour
2	High	8 hours
3	Medium	24 hours
4	Low	48 hours
5	Planning	Scheduled

The target resolution times need to be carefully thought out to correspond with agreed service targets. Automatic escalation times can then be set, based upon these target resolution times. No target resolution time is shown for major incidents, as by their very nature, these incidents are above and beyond what is considered “normal,” and hence resolution times will vary considerably. Escalation during major incidents is the responsibility of the major incident manager.

Initial Support

Initial support is the process whereby the service desk is able to resolve the incident to the satisfaction of the customer without reference to any other support areas. The service desk may be able to achieve this by matching the reported incident to a known error and providing details of the known resolution or workaround to the customer. In this case, the incident record must be linked to the known error record so that all affected users can be identified when a permanent resolution is available and to assist when reviewing the priority for resolving known errors.

Alternatively, the service desk analyst may be able to supply resolution details from personal experience or expertise gained through a defined training program. When neither of these is possible, the service desk analyst might need to search knowledge bases to match the symptoms described by the customer. If a

resolution is proposed by either source, then the known error database should be updated and full details of the resolution should be recorded against the incident record. It may be appropriate to advise the configuration manager so that the CMDB can be updated, too.

A known error is a record of a previously reported and resolved incident. There will be at least a workaround available, if not a complete solution. Where a known error database exists, it should be available to all service desk analysts and must be maintained carefully. It might be the case that a workaround is applied temporarily while a complete solution is being developed or while awaiting release of a new version of software. This may take some time and the error might affect many users. Using the details recorded in the SLA, the service desk can identify the business users at risk and adopt a proactive approach, giving warning of the potential problem and advising of the existence of a workaround.

Technology note: Knowledge bases may be developed internally by an organization, perhaps using database or specialist CRM tools, or be provided by suppliers using browser technology. If internally developed, then the design considerations should include definition of mandatory fields, key word usage, and search methodology. A single point of ownership should be established to control content and visitor tracking should be incorporated so that usage can be monitored. As the facility develops, it will be used more frequently, and it is important to track the extent and scope of usage, as this provides useful data for trend analysis.

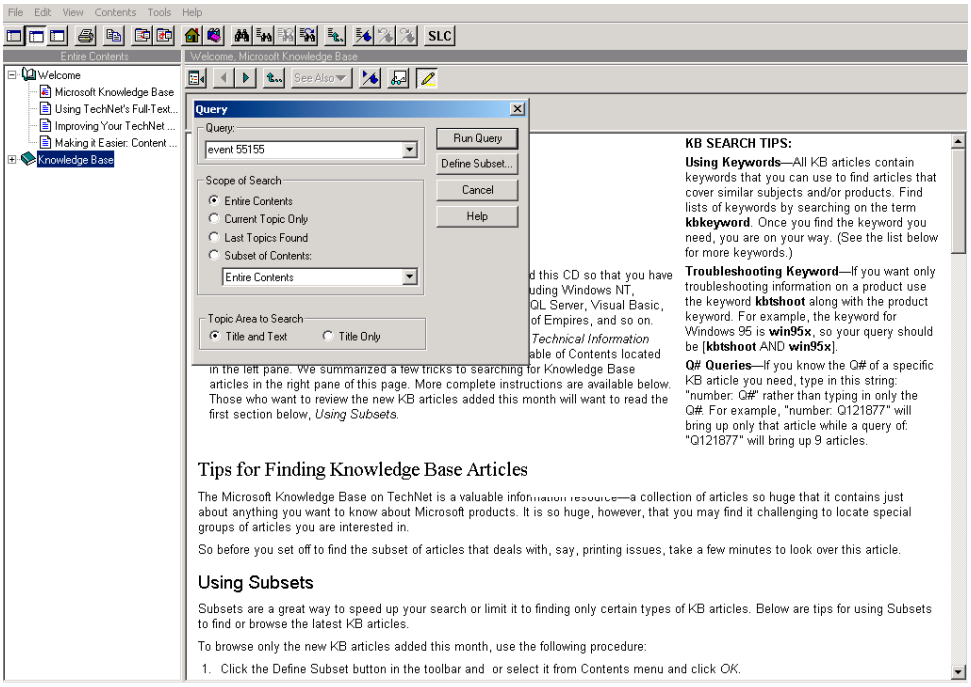


Figure 13

Example of TechNet, a knowledge base available from Microsoft

If reliance is placed on supplier-provided data, then there might be constraints on these criteria, although it might be possible to influence content.

User group information must be treated with caution, as it is likely to include contributions relevant to a range of release versions, and there is little control over terminology, structure, or content. User group information is often based on bulletin board or discussion group methodology but can be a good source of current experience. Their use is best restricted to service desk and resolver groups rather than being incorporated into a self-service environment.

Implementing a workaround is a perfectly acceptable way of dealing with an incident. A workaround might be a temporary response while a more permanent solution is being developed or it might be the permanent solution itself.

If the workaround resolves the incident, then the service desk analyst can proceed to take incident closure action. If it does not succeed in this instance but has been successful previously, then the incident was incorrectly matched and must be unlinked and returned to initial support for matching against existing problems as the next stage in the incident management process.

If the incident can be matched to an existing problem, a similar process is followed to that for a known error. If a match is found, then the incident must be linked to the problem record and, if a workaround exists, then the workaround must be implemented, resolution confirmed, and closure action taken. If resolution is not achieved, then the incident must be unlinked from the problem and returned for further investigation.

If the incident has been linked to an existing problem, but problem management is still working to identify a workaround, then the initiator should be informed of the problem record reference and the status. Problem management should then continue working on the issue and inform incident management when a workaround or solution has identified.

Incidents failing to match known error or existing problem records can be compared to previous incidents. Historical incident records are frequently a good source of information, providing details of how similar incidents were resolved in the past. This is where the need for incident records to be fully updated, with clear concise descriptions of symptoms and actions taken, becomes paramount.

If an incident matches one or more previous incidents, then it may be an indication that an underlying problem exists. If there is any doubt, problem management should be advised. It is then up to problem management to decide whether to create a new problem record.

Technology note: The matching of incidents, problems, and errors can be most effectively performed if an integrated tool is in use. Even when an integrated tool is deployed, the benefits can be lost if the tool is not used consistently throughout the support organization. If incidents, problems, and known errors are recorded and updated in a controlled and consistent manner, then the resulting data will allow far more productive matching and reporting. It is vital that ownership of the tool be maintained and that control be exercised to ensure that all parties accessing the tool use it in a consistent manner as defined within documented procedures. Ideally, this consistent and controlled use of the tool should be in place when it is first deployed, as it is often very difficult to re-establish usage policies once teams are each using the tool to their own individual standards.

If an organization uses local service desks, then matching can often only be done locally, limiting the potential for sharing the accumulated knowledge that is being built up within the tool sets. Global organizations will experience more difficulty achieving integration, as cost, language, and infrastructure considerations may make use of a single tool difficult, or perhaps impossible. Additionally, some business processes may be concentrated in specific countries and be of limited general interest, making integration less valuable.

Initial support staff might be able to resolve incidents by using their own implicit knowledge built up over time. Organizations may increase the level of implicit knowledge available on the service desk by offering business staff the opportunity to move into the support organization, bringing their experience of the business processes and applications with them. Formal training can also be used to increase the proportion of incidents being resolved at first contact. However, time, as well as knowledge, is an issue here. Frequently, initial support staff may feel that they have the knowledge to delve further into a particular incident, but the incident has to be passed to a resolver group due to the pressure of incoming calls. Deciding how much time should be spent on the initial support of an incident is a key capacity issue for the service desk and has implications throughout the support chain.

Where possible, staff should attempt to resolve incidents during initial support by providing resolution advice, as long as this does not impact targets for the answering and handling of incoming calls. “First call fix success” is a valuable metric and a key component of customer satisfaction and cost reduction. When resolution is thought to have been achieved, incidents should be progressed to the closure process.

In cases where initial support has failed to resolve the incident, the incident should be assigned to a resolver group based on the incident classification. Depending on the type of incident, the resolver group may be an internal team or an external vendor. In either case, the service desk retains ownership of the incident and is responsible for monitoring and tracking progress of the incident.

The initial support team needs to have access to details of any enterprise support agreements, including contracted service levels, contact details, service hours, and expected response times. It is vital when incidents are being escalated to external vendors, that the

support staff contacting the vendor has all the required details. For each external contract, a list of the information that will be requested when logging a call needs to be produced and maintained. This information typically includes the contract or agreement number, a site ID, details of the products affected (hardware model and serial numbers, or software versions and patch levels), and advice on the level of priority needed from the vendor. Some contracts may only allow new calls to be logged by named contacts. If this is the case, then the list of named contacts must be up-to-date and readily available.

Technology note: Interfaces for passing incidents to external vendors are more integrated than having to telephone or fax the vendor. The two most common scenarios are:

- External vendors have their own queue within the organization's service desk tool and have access to update and resolve incident records. It is the vendor's responsibility to monitor their queue and progress any incidents assigned to it.
- The organization's support desk tool has an interface with that of the external vendor. When an incident is assigned to the vendor's queue, a corresponding incident record is recorded within the vendor's tool. Depending on the level of integration, the original incident record is automatically updated when the vendor's record is updated (or at regular intervals), or the organization has read access to the vendor's self-service facility for obtaining progress updates.

The key factors in successfully achieving this level of integration are:

- The interface needs to be as automated as possible and not cumbersome for support staff to use.
- The interface needs to have the correct levels of availability and reliability.
- The interface needs to provide both parties with appropriate security.
- Responsibilities for monitoring and progressing incidents need to be clearly documented and understood.
- Target resolution times and priorities need to map well between the organizations and be clearly understood.
- Responsibilities for maintaining the interface need to be understood, including an awareness of the impact that changes to one support tool will have on the other organization's tool and processes.

Investigation and Diagnosis

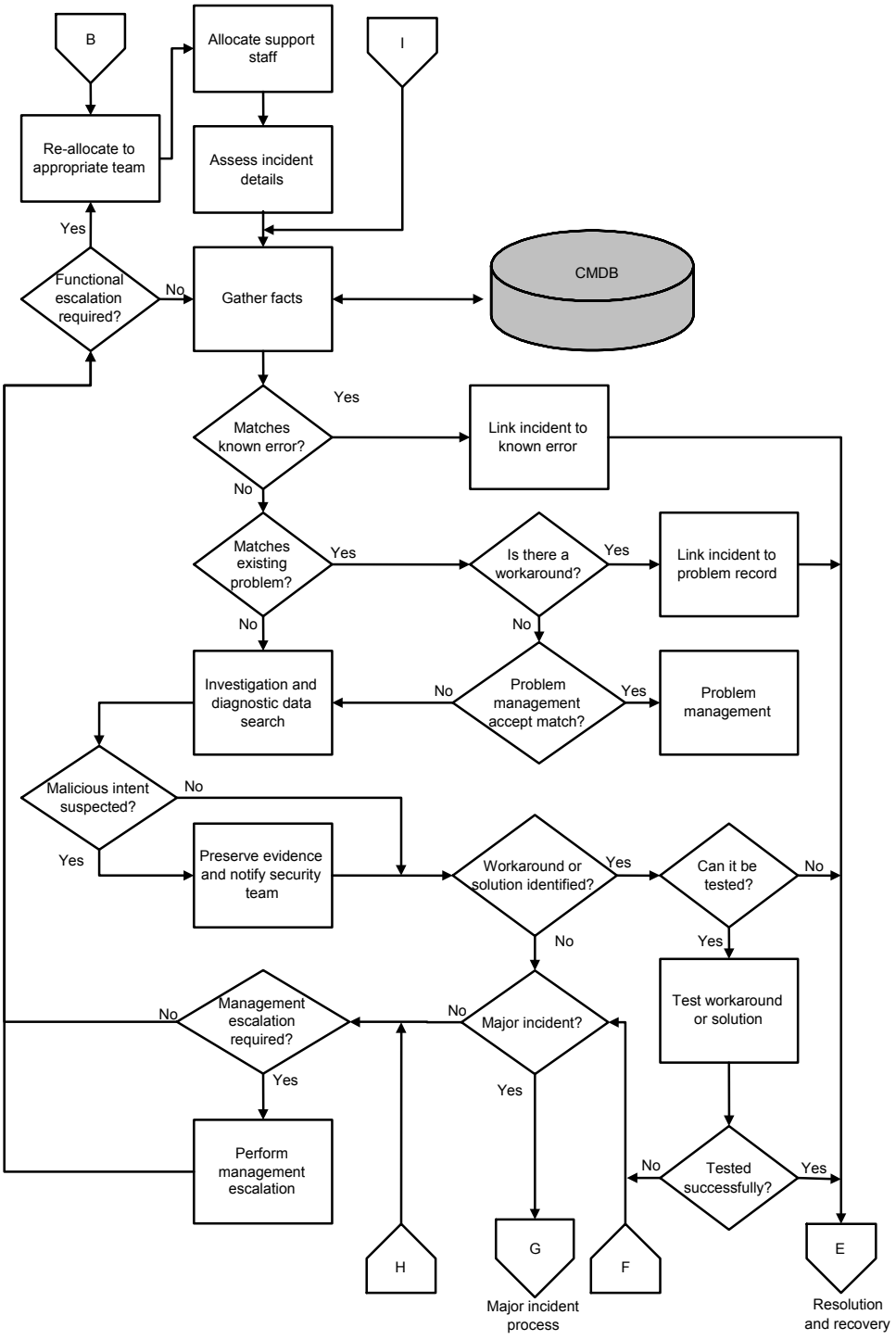


Figure 14

Investigation and diagnosis process flow chart

If initial support has been unable to resolve an incident, it passes to the investigation and diagnosis process. This is initiated when the incident is assigned to a resolver group. Resolver groups may include a wide range of IT teams, including support and development

personnel, other SMFs, other units within the organization, outsourcing providers, partners, and other third parties.

The number and size of resolver groups depends on the organization. Typically, support organizations are structured using one of the following approaches:

- A tiered structure comprising first-line, second-line, third-line and n^{th}-line support teams. Often n^{th}-line will be either an external vendor or an in-house development team, depending on the applications supported and the level of in-house expertise.
- A platform- and application-based structure consisting of platform-specific support teams, often with additional teams focused on database and messaging applications.
- A combination of the above structures. The most common is a first-line support team (at the service desk), backed by a platform-based structure of support teams consisting of staff with a variety of skill levels, supported by external vendors providing n^{th}-line support for their own products or complex/major incidents. Some service desks operate a front office/back office structure, in effect creating two tiers of support within the service desk itself.

The structure needs to meet the needs of the individual organization, but it should be remembered that increased complexity brings increased cost. The higher the number of individual support teams, the more difficult it is to share information and provide skill duplication—and the more likely it is that arbitration will be required to decide responsibilities when dealing with complex incidents. In instances where responsibility for resolving an incident is in dispute, the service desk should discuss the issue with the parties involved to see if an agreement can be reached. Where no easy agreement can be reached, problem management should be responsible for arbitration.

Whatever the number and complexity of the resolver groups, the structure must be clearly documented so that all staff understand which categories of calls should be assigned to each team and what the escalation paths are, if required. The lack of a clearly defined support structure results in increased cost, frustration, and resolution times, while decreasing customer satisfaction.

The service desk tool should have a queue for each resolver group. It is the resolver group's responsibility to monitor its queue and to progress incidents assigned to it. Support staff members may log on to the service desk tool and view a regularly refreshed list of incidents within their queue.

Technology note: A wide variety of technology is available to assist resolver groups with their queue management. These include displaying the queue status within the resolver group's office area by using large screen monitors or purposely designed electronic display boards. Large electronic display boards can be configured to display the key outputs from a number of different systems, including service desk systems and event management systems.

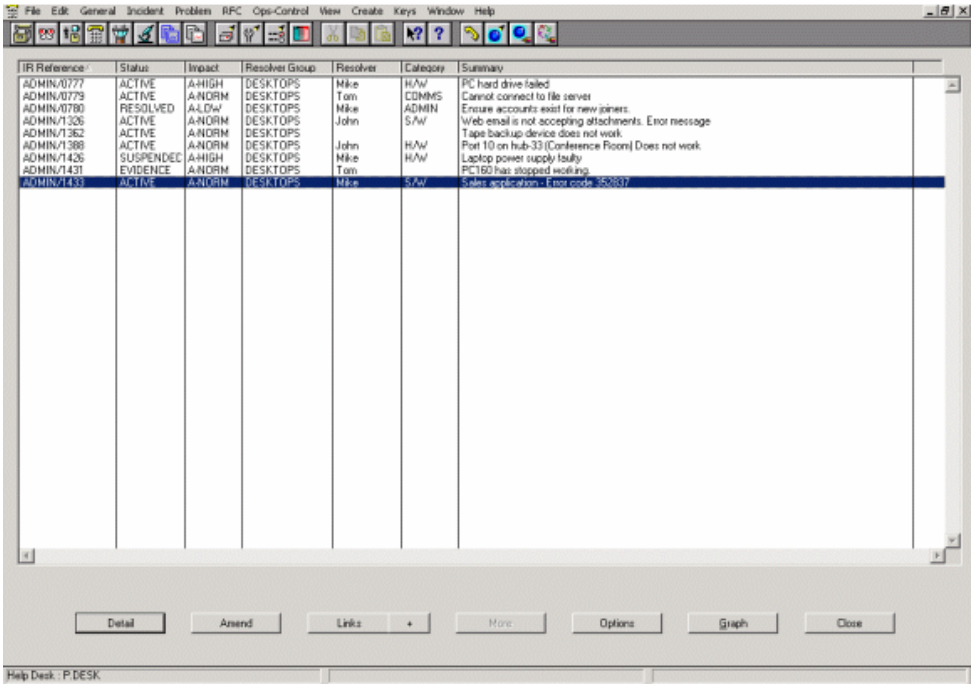


Figure 15

Example of a resolver group queue

Many support staff are mobile and need to be able to see their queue from various locations. This can be achieved by allowing support staff to connect to the service desk tool via an internet browser from any personal computer within the organization. Wireless technologies allow support staff to be equipped with personal digital assistants (PDAs), which can be synchronized with the service desk system to display an up-to-date queue as well as to update existing incidents. Steps should be taken to ensure the security of these systems.

Where staff cannot be constantly monitoring the queue, technology can again be employed to provide them with notifications when new incidents are assigned to the queue. Notifications may take the form of pop-up boxes, e-mails, or mobile text messages.

When an incident is placed within a resolver group queue, it should then be assigned to a team member to progress. It is the team member's responsibility to maintain the incident record, updating it with the latest status, actions taken, and progress.

Incident records should be updated with the following information for each action taken:

- Name/id of the support group and person recording the action.
- Type of action (re-assigning, diagnosis, recovery, resolution, closure, and so forth).

- Date/time of action.
- Description and outcome of action.

If at any stage it is felt that the incident needs to be assigned to a different resolver group, then the current resolver group should re-assign the incident back to the service desk, which will then make the new assignment. This reduces the likelihood of calls being continually bounced between resolver groups and allows the service desk to more effectively coordinate the incident management process by spotting situations where arbitration is required.

Resolver groups should not be allowed to close incidents themselves. Once an incident is thought to be resolved, it should be put into a status of resolved and assigned to the service desk. The service desk analysts should then confirm that the initiator is happy with the resolution prior to closure.

Once an incident has been assigned to a staff member within a resolver group, he or she should assess the details of the incident up to this point and gather any required facts.

Technology note: The configuration management database (CMDB) can be an important tool for fact finding during incident investigation. The database can be used to view details of the configuration items (CIs) affected, such as the software versions loaded on them and the history of these CIs, and to check the details of any support contracts or warranties that cover the CIs, confirming their locations.

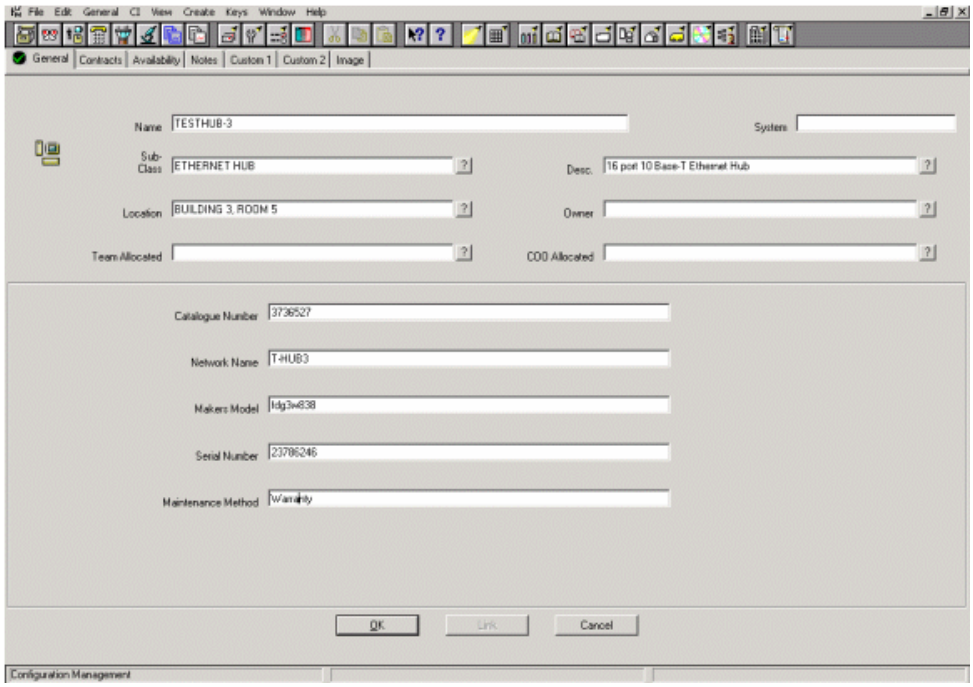


Figure 16

Example of a basic configuration item record

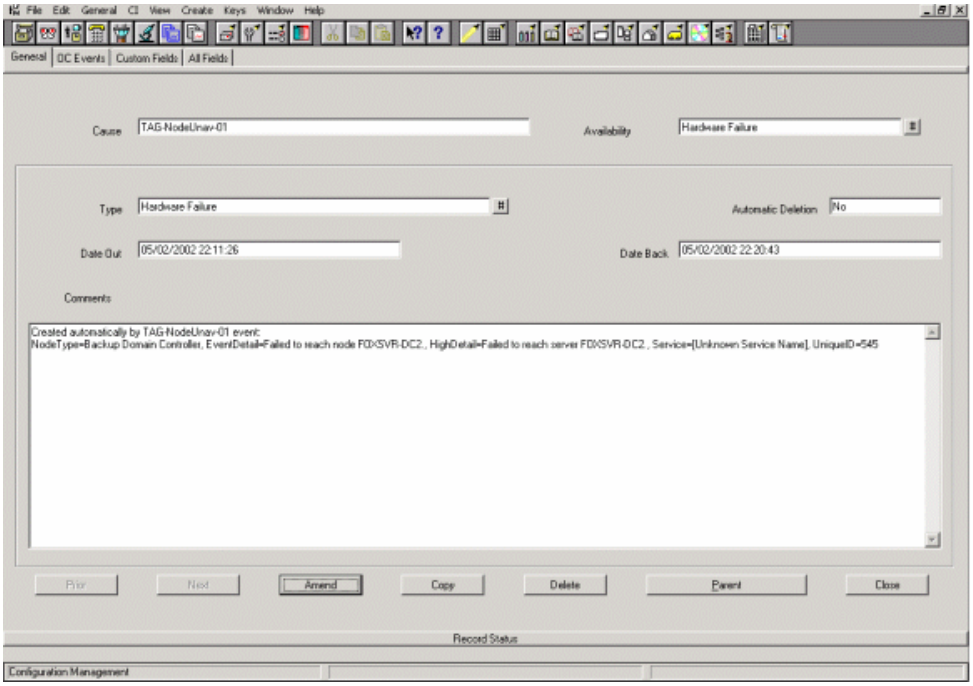


Figure 17

Example of a configuration item history record

The CMDB can also be used to identify CIs that are related to the affected CIs, so that steps can be taken to minimize the impact of the incident by re-routing or re-configuring the related CIs.

Another use of the CMDB is to identify other CIs that are identical to the failed CIs, so that they can be used for comparison purposes, hopefully highlighting the differences that are causing the incident.

Once the support staff has completed any fact gathering, they should consider whether the information gathered so far indicates that this is an occurrence of a known error. Known error information may come from a variety of sources—for example, corporate known error databases maintained by problem management, vendor-supplied known error databases, and Internet sites, including vendor sites and related newsgroups.

Technology note: The wealth of information available within vendor-supplied knowledge bases makes them a vital tool for support teams. These knowledge bases are available either on vendor Web sites or as CD/DVD packages, often purchasable with either site or individual licenses.

The knowledge bases contain a variety of information, including known errors, fixes, updated drivers, white papers, frequently asked questions (FAQs), and technical articles.

The great advantage of vendor-supplied knowledge is that organizations can benefit from the diagnostic and resolution information accumulated when incidents have occurred at other sites, often resulting in a much quicker fix time if the organization encounters the same or similar incidents.

Vendors also often supply considerable help information within their products in the form of online help documentation and search facilities. In some cases, additional documentation and useful utilities can be obtained from product-specific resource kits.

Some systems management tools also contain knowledge bases so that when an alert is generated, additional diagnostic and/or resolution advice can be supplied. As with all vendor-supplied knowledge bases, mechanisms need to be in place to ensure that regular updates are obtained and applied in order for the supplied information on known workarounds and fixes be up-to-date.

Organizations may also build up their own corporate knowledge bases. These corporate knowledge bases can contain any information that is relevant to staff supporting the organization's IT infrastructure. Whereas vendor supplied knowledge bases may contain a large amount of material that is not relevant to a particular organization, the corporate knowledge base can be constructed to contain information specific to its operations. Corporate knowledge bases can be constructed by using in-house developed databases or third-party tools designed for the purpose.

If the incident does match an existing known error, then the incident record should be updated to associate it with the known error record. Depending on the tools used, this may involve linking the two records in an integrated tool set. However, with separate tools, it could just mean including the known error ID (and whatever additional detail space allows) within the incident record and, if possible, updating an incident count on the known error record. The incident should then progress to the resolution and recovery process, where the workaround or solution identified within the known error record can be applied.

Incidents that cannot be matched to known errors should then be checked against existing problems. Existing problems are problems that are currently known to problem management, but where a resolution for the root cause has yet to be identified. An existing problem may or may not have an identified workaround. If the incident matches an existing problem and there is an identified workaround, then the incident should be associated with the problem record (by linking or cross-referencing) and then progressed to the resolution and recovery process for application of the workaround.

If the incident is thought to match an existing problem but no workaround has yet been identified, then the incident record should be passed to problem management. It is problem management's responsibility to confirm whether the incident does match the

problem and either accept the incident (associating it with the relevant problem record) or reject it back to incident management if the match is found to be incorrect. Problem management continues working on existing problems and is responsible for informing incident management when either workarounds or solutions are found to problems that have associated incident records. Once informed, it is then incident management's responsibility to resolve, recover, and close any outstanding incidents.

The process of matching incidents against known errors and problems continues throughout the investigation and diagnosis of the incident. As further diagnostic evidence is collected—for example, new event IDs or error codes—this new information should be fed into the matching process. As part of the investigation process, previous incident records may also be reviewed to establish how any similar incidents have been resolved in the past.

Incident management is usually focused on restoring normal service as quickly as possible. The exception to this is in cases where malicious intent is suspected. In these cases, the support staff should work alongside security staff and in accordance with the security team's policy and procedures for handling security incidents. It is often critical to ensure that all available evidence is preserved for use during legal or disciplinary proceedings. The security team should assist with the collection and storage of evidence, providing guidance on which evidence will be admissible and what context details (computer IDs, date and time stamps) need to be included. Depending on the impact caused by the incident, a decision may need to be made whether to resolve the incident on the live system, possibly overwriting vital evidence, or to switch over to standby continuity arrangements until it is sure that all available evidence has been collected.

The investigation and diagnosis phase involves support staff first getting a clear picture of how normal service is being impacted, then identifying what is causing this impact, and finally determining how it can be worked around or resolved.

When an incident is first reported, there may be very limited information on the nature, extent, and overall impact of the issue. The incident record should include as much detail as possible from the initiator, but often this information will be limited to the individual's viewpoint and degree of knowledge. In order for support staff to resolve the incident, they must first find out more about it. This should include finding out the exact error messages or event IDs produced, confirming any actions taken leading up to the incident, and confirming the scope of the incident—for example, does it affect just the single user who reported the incident or all users of the service. Support staff should gather evidence by looking in log files

and event logs and, if feasible, trying to replicate the issue so that they can observe it first hand.

How easy it is to actually visit the user to observe the issue varies depending on geographical location. As visiting even local users takes time, the aim should be to fix as many incidents as possible remotely. As previously stated, the goal of incident management is to restore normal service as quickly as possible, so if this can be done remotely without the need for transit time, this is in line with the goal. However, if at any stage during the incident life cycle it becomes likely that a visit, despite the transit time, would bring about a quicker resolution, then this should be considered. Part of the consideration should include justifying the costs involved with a visit against the impact of taking longer to resolve the incident. While policy should be to minimize the overhead of time spent in transit, organizations need to ensure that this does not prevent justifiable visits and thus prolong incidents.

Consideration should be given to the structure of the support organization and whether support staff should be centrally located or distributed across sites. Generally, distribution adds complexity and hence cost to the support organization. However, frequently the structure has to be designed in order to meet agreed service levels. For example, if the agreed service level calls for a one-hour onsite response and the site is distant from other support locations, then some form of local presence would be required. Depending on the costs involved, the security implications, ease of recruitment, and the level of support required, organizations may consider outsourcing the onsite support of far-flung locations.

Technology note: Remote support tools are available to help minimize the need for onsite visits. These tools include utilities that allow logs to be viewed remotely and administrative utilities to be run against remote computers. Steps should always be taken to ensure the security of these systems against unauthorized use.

Increased functionality can be had by using remote control tools that allow support staff to view what the user is experiencing on a remote system and, if necessary, take over the system to investigate the incident and apply any resolution actions. These tools are also essential when support staff is responsible for systems located in remote or unattended computer rooms.

Once the support staff understands the incident, they then need to identify why it is occurring and how to either resolve the issue or work around it. Incident management is interested in the immediate cause of the incident rather than the possible underlying root causes that problem management is interested in. An example of this is that incident management might identify the cause of an incident as a file server that has "locked up" or "frozen" and reboot it to resolve the incident, while it would be up to problem management to determine whether this was an isolated incident or whether there is an underlying problem that is causing this (and possibly other) file servers to lock up.

Despite thorough change and release management processes, it is likely that some incidents will still occur because of changes that have been applied. When an incident occurs and no immediate cause can be identified, support staff should check what changes have recently occurred to see whether any might be relevant. Formal change management not only aims to prevent further incidents from occurring because of changes, but also ensures that when such incidents do occur, full details are available about the changes that have been made.

When troubleshooting complex incidents, support staff should study all available evidence and then break down the incident into simpler units. To ensure an organized and logical approach, it is useful to present the issue as a graphical representation by drawing it on paper or a whiteboard or by using a software tool. It can be in the form of a flow chart, data flow, or network diagram. A suitable graphical representation can help identify the components and interfaces involved, and hence the potential failure points. A brainstorming session may be required at this point to identify all of the possible failure points. If several potential failure points exist, a weighting can be applied that evaluates the perceived likelihood of each failure point being the immediate cause of the incident and the ease of checking each failure point. The weighting can then allow investigative actions to be prioritized.

In order to allow support staff to replicate incidents and test workarounds away from the live environment, they should have access to test environments that mirror the live environment (or

portions thereof) as closely as possible. While it is rarely possible to exactly replicate the size and workload of the live environment, significant benefits can still be obtained from a scaled-down test environment. Justifying the cost of test environments can be a difficult task. However, this investment should be made in order to prevent a constant cycle where changes made (using change and release management processes) to resolve one incident lead to another incident because inadequate testing was carried out prior to making the change in the live environment.

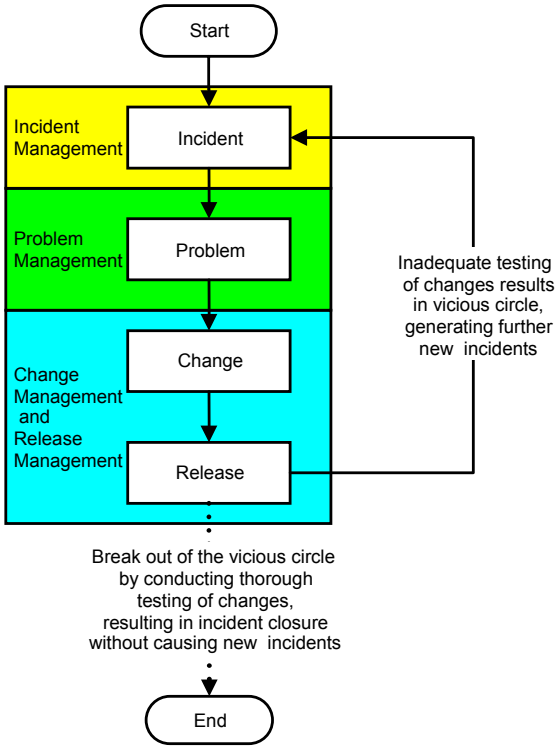


Figure 18

Example incident-change life cycle

To assist with the diagnosis of incidents, support teams can use techniques such as mentoring and triage sessions.

Mentoring can be provided to junior support staff by more experienced individuals, possibly from higher tiers within the support structure. The more experienced staff can provide advice and guidance on how to investigate a specific incident and how to resolve it, once the incident has been diagnosed. This has the benefit of increasing the knowledge and experience of the junior staff, while also reducing the number of incidents needing to be assigned to more experienced staff, thereby freeing up more of their time to handle more complex and strategic issues. This process of knowledge sharing should be encouraged. However, incidents should still be escalated when the staff feels it is appropriate in order to meet service targets.

Regular informal triage sessions can be held with team members. During these sessions, the details of any outstanding incidents can be discussed so that other team members can suggest things to try and to offer the benefit of their own experiences. Again, this is a form of knowledge sharing that utilizes team members' implicit knowledge of the infrastructure and environment. Where teams are geographically diverse, teleconferencing or video conferencing facilities can be used for these meetings.

A different form of knowledge sharing sometimes employed is to use topic-specific e-mail lists to raise questions. When an individual has a question regarding a specific topic, he or she can send an e-mail to the relevant distribution list to see if anyone on the list can offer advice or a solution. For example, there could be an e-mail list that includes all the staff with database knowledge, another for messaging products, and another for base operating system knowledge.

Depending on the incident being investigated and the experience within the organization, it may be appropriate to seek knowledge externally by posting a question on a relevant Internet-based forum. Many forums are available and are often aimed at a specific platform or application, which provides an opportunity to discuss difficulties and share knowledge with support staff in other organizations who are working on the same products.

Technology note: Some service desk tools allow electronic evidence such as log files, event records, or job journals to be attached to the incident records. Staff working on the incident can then open up the attached documents directly from the incident record, rather than having to search for them.

Once a possible workaround or solution has been identified, this should be tested away from the live environment if possible. In the past, it has been an all too frequent occurrence that changes implemented to resolve one incident have resulted in a number of other incidents. To avoid this, support staff should test proposed workarounds or solutions as thoroughly as possible and then hand over any required configuration item (CI) changes for implementation via the change and release management processes.

When testing shows that a proposed resolution will not work, the investigation and diagnosis process should continue until a successful resolution is identified. If testing is successful, then the resolution can be progressed within the resolution and recovery process.

At any time during the investigation and diagnosis of an incident, it may be necessary to declare a major incident. Major incidents are incidents with a high impact, or potentially high impact, which require a response that is above and beyond that given to "normal" incidents. Typically, these incidents require cross-company coordination, management escalation, the mobilization of additional

resources, and increased communications. Although an incident might be suspected to be "major" when first detected, the support staff needs to confirm the scope and impact of the incident before the major incident procedure is initiated. Incidents suspected as being "major" should be logged with a high priority and the relevant resolver group notified so that they can confirm the scope and impact as quickly as possible.

Organizations should document criteria by which they identify major incidents—for example, a retail bank might decide that any incident affecting normal service at 20 or more branches should be classified as a major incident. Suspected major incidents should then be flagged to the duty incident manager, who is responsible for deciding whether to initiate the major incident procedure, based upon the evidence and following consultation with relevant parties.

During major incidents, the process of investigation, diagnosis, resolution, recovery, and then closure still continues. However, the major incident procedure oversees these activities and provides increased coordination, resources, and communication.

If an incident is not a major incident, it may still need some form of escalation. Escalation is a mechanism that assists the resolution of an incident within the agreed service targets. There are two forms of escalation: management, or hierarchical, escalation and functional escalation.

Management, or hierarchical, escalation may be carried out at any stage during the incident life cycle if it is thought that the incident will not be resolved satisfactorily or in time. It is the responsibility of staff within the service desk and resolver groups to escalate the incident to management as soon as an unsatisfactory or untimely resolution becomes likely. Escalation should be initiated sooner rather than later so that there is still time for management to assess the situation and implement a corrective action. Corrective actions might be to allocate additional resources or to seek specialized skills from elsewhere, either within or external to the organization.

The need for functional escalation should also be considered throughout the life of an incident. Functional escalation concerns transferring an incident to different support staff, who are better equipped to progress the incident and achieve a resolution within the agreed service targets. In a tiered support structure, this could involve transferring the incident from second- to third-line teams; while in a platform-based structure, this could be an allocation to more experienced staff within the team or allocation to a different team because the incident category is different from that first thought. Functional escalation also includes raising the incident with external support resources and vendors.

All escalation actions should be recorded on the incident record.

Technology note: Service desk tools allow automatic escalation to occur based on time spent or time remaining until service targets are reached. While escalation times need to be carefully planned so that service level targets are not exceeded, lower tiers of support need to be given a reasonable amount of time to resolve incidents so as to reduce the impact on more expensive, "higher tier" resources.

Organizations may decide to implement automatic management and functional escalation based on time factors. The automatic actions might in many cases simply involve notifying the relevant people that escalation is now due on a particular incident, so that they can take appropriate actions. Tools may allow the "escalation clock" to be stopped when the incident is in certain states—for example, "waiting for evidence," where the support staff have asked for further information and are waiting for the contact to supply this, or "resolved," where the incident is believed to be resolved but the service desk analysts are waiting for confirmation of this from the initiator prior to closing the call. In these instances where progress is waiting on the initiator, the "escalation clock" may be stopped. However, the incident manager should monitor to make sure that support staff are not "playing the system" by requesting further, difficult-to-get evidence, so as to increase the time they have to investigate the problem.

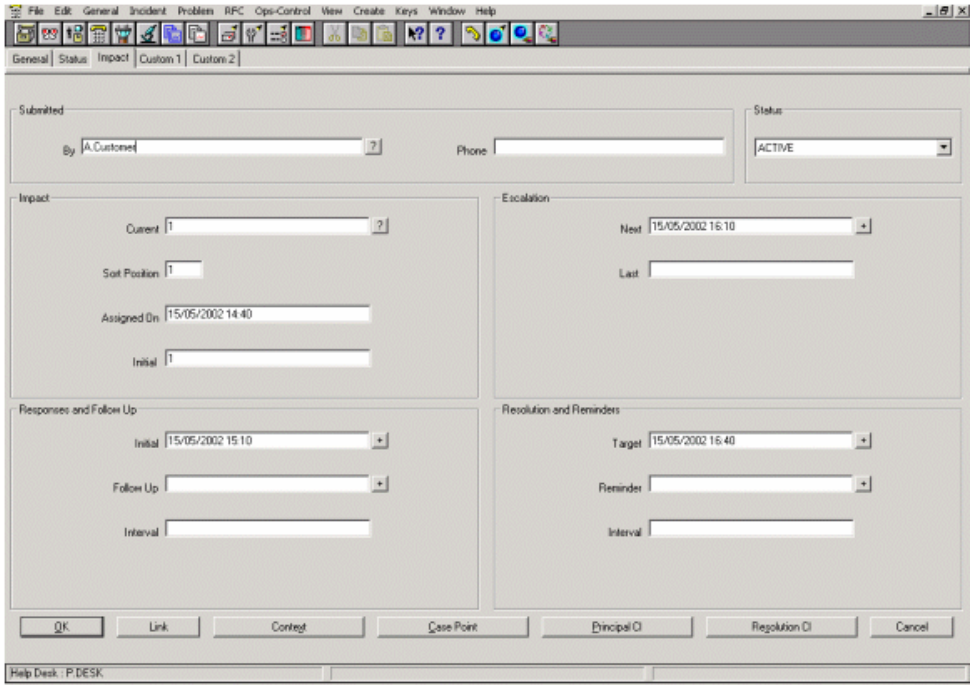


Figure 19

Example of incident record showing escalation and target resolution times

Major Incident Procedure

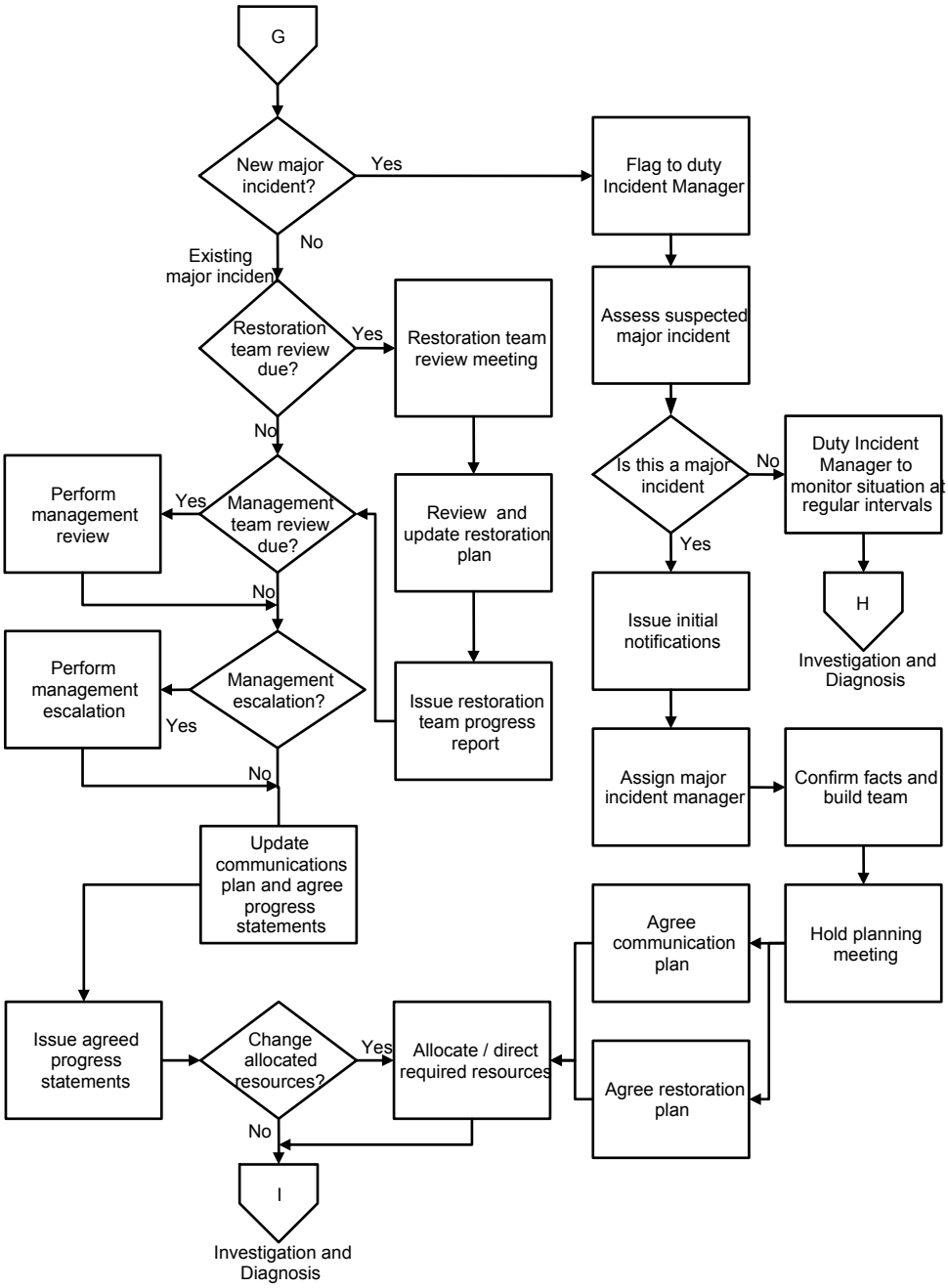


Figure 20

Major incident procedure flow chart

The major incident procedure provides additional cross-company coordination, resources, management involvement, and communication in instances where the impact, or potential impact, of an incident necessitates a response that is beyond that provided by the normal incident management procedures.

The decision on whether an incident warrants the invocation of the major incident procedure rests with the duty incident manager,

although consultation should occur with all parties involved before the decision is made. Support staff, service managers, business managers, partners, and other IT managers involved in the incident may need to be contacted, to ensure that the duty incident manager fully understands the situation and the potential implications.

The organization should document a number of criteria to act as guidelines in deciding what should be considered a major incident. Criteria will be specific to the business but could involve:

Financial impact. Any incident where the potential impact could exceed a stated figure.

Service- based. Any downtime of key business-critical services, where a quick resolution does not appear likely.

Site- or user-based. Any incident affecting more than a specified number of sites or users, where a quick resolution does not appear likely.

Target-based. Any incident that is likely to cause specific service level targets to be breached, especially where significant financial penalties could be incurred.

Health- and safety-based. Any incident where it is felt that health and safety will be directly at risk until a resolution can be achieved.

Security-based. Any security incident where it is felt that either a major loss or impact has been sustained, or where a major vulnerability has been exposed.

Reputation-based. Any incident that needs to be responded to very quickly and/or effectively in order to maintain or minimize impact to either the organization's or a brand's reputation.

The criteria need to ensure that the major incident procedure is invoked when threatened with significant business impact, while preventing too many unnecessary and costly invocations where the situations do not warrant this level of response. These criteria should only be used for guidance; it is always up to the duty incident manager's judgment whether to invoke the process. The duty incident manager should base the decision on a flexible, common sense approach, considering the individual characteristics of the incident and ensuring that the criteria for labeling an incident as major are not too rigorous. Invoking the major incident procedure will normally have an impact in terms of disruption and the cost of mobilizing extra resources, so care needs to be taken to ensure that it is used only when strictly necessary.

Anyone within the organization should be able to contact the duty incident manager regarding an incident that he or she thinks should

now be considered a major incident. Although in many cases this contact will come from the service desk or a resolver group, there should be nothing to prevent other staff such as service managers or business managers, who often have a greater understanding of the potential impacts, from contacting the incident manager. To ensure that this is possible, the major incident procedure needs to be publicized along with the means of contacting the duty incident manager.

If the incident manager decides that an incident does not warrant invoking the major incident procedure at the current time, then the reasons need to be explained to the parties involved, and then the situation needs to be monitored while the incident is progressed normally. Once an incident has been flagged to the duty incident manager, even if it is not currently considered a major incident, the incident manager should still continue to monitor the incident regularly, in case the situation changes.

Once an incident has been identified as a major incident, the incident manager should immediately notify all interested parties: problem management, availability management, service level management, service continuity management, IT management, affected business managers, and the duty service desk manager.

Someone must assume the role of major incident manager, responsible for coordinating the planning, resources, and communications during the major incident. Depending on the size of the organization, the potential threats and impacts, and the number of major incidents experienced, this may be a permanent position or one that is assumed by appropriate staff when required. In services organizations providing support for multiple customers, there may be a requirement for a number of permanent major incident managers. If an incident is to be worked on around the clock, then more than one major incident manager may need to be assigned to a single incident.

If the role of major incident manager is to be a part-time role, then depending on the organization, this could be assumed by a problem manger, availability manager, or service manager.

Staff performing the role of major incident manager needs to meet the following criteria:

- Able to handle the stress generated during major incidents.
- A senior manager with the authority to make things happen.
- A good communicator, able to talk to technical IT staff, business representatives, and customers at all levels within the organization.
- A recognized figure with an in-depth knowledge of the organization, including how things work and who to speak to.

- Prepared to work extended hours, often at short notice.
- Prepared to travel and to be present on customer sites if required, again often at short notice.

Once a major incident manager has been assigned, he or she should gather all the information so far available and confirm the current situation. The major incident manager is then responsible for forming the restoration team.

The technical staff involved in investigating and resolving a major incident is referred to as the restoration team. This team typically consists of one or more technical staff. Depending on the size and nature of the organization, there may be technical staff permanently "on the bench" waiting to respond to major incidents. If so, these should be experienced staff members who are trained in both technical skills and problem solving techniques. This core team should be trained in all of the key strategic technologies in use, but be supplemented by other support staff if the particular incident calls for a different skills profile. Experienced support staff should be rotated through the core team on a temporary assignment basis.

If resources do not allow an "on the bench" core team to be maintained or if the number and likelihood of major incidents does not warrant this investment, then an "as-needed" restoration team needs to be formed by the major incident manager when a major incident is recognized.

The major incident manager should hold an initial planning meeting with the restoration team, any staff members who have already been working on the incident, affected managers, and any other relevant technical specialists. If the staff is geographically diverse, this meeting may take the form of a teleconference or video conference. The objective of the meeting should be to agree both a restoration plan and a communications plan.

The objective of the restoration plan is to provide a planned and coordinated approach for providing restoration of service. The plan should be owned by the major incident manager and should document the actions that need to be taken, who should carry them out, and when they should be completed. The plan should be regularly updated throughout the life of the major incident, ensuring that old versions are kept as an audit trail.

The restoration plan should contain the following:

- Statement of the "problem" as known at this time.
- Breakdown of the incident detailing components, interfaces, and likely causes of the difficulty.
- High-level plan of how to verify or rule out each possible immediate cause.

- Weighting of each possible cause based on likelihood and ease of confirmation, allowing the investigation of each possible cause to be given a priority.
- Details of which investigative or resolving actions will be taken at this stage, based on the assigned priorities.
- Details of who will carry out the investigative or resolving actions.
- The timescales that each action should be carried out in, and the time of the next restoration team review meeting.

An example restoration plan template is included in Appendix E of this document.

The major incident manager should review progress regularly with a management team consisting of IT managers, affected business managers, and partner and customer management. The purpose of the management team meeting should be to keep all parties informed, discuss progress, and provide management escalation when required. Once a management review meeting has been held, progress update statements should be agreed and issued.

Management team and restoration team review meetings should normally be kept separate, so that each team can concentrate on the issues relevant to their roles. The restoration team should discuss the technical issues and then provide a progress report for the management team, who can then concentrate on resources, escalation, and communication issues. Keeping these meetings separate minimizes the time that individuals spend in meetings (and not actually working on the incident), while also allowing each group to focus their discussions.

During major incidents, the handling of communications can often become a major difficulty in its own right. The objective of the communications plan is to provide coordination of all communications during the life of the incident. The communications plan should be owned by the major incident manager and be discussed and updated at each management team review meeting. The plan should cover:

- Who needs to be regularly updated.
- Contact details for all parties requiring updates.
- The different types of updates that will be required. Different update messages may be required depending on the audience receiving the communication:
    - Senior management update
    - Update for all staff
    - Update for customers
    - Update for partners

  - Update for staff working on the major incident
  - Press/media statement
  - Update for emergency services/authorities
- How often each type of update is required and when the next one is due.
- Who is authorized to agree the release of each different update statement.
- The mechanism by which each update will be communicated.
- The time of the next management team meeting.

A template communication plan and suitable e-mail distribution lists should be developed for use during major incidents; however, consideration must also be given to alternative methods of communication in the event of specific mediums, such as e-mail, being unavailable.

An example communications plan matrix is included in Appendix F of this document.

Once a communications plan and a restoration plan have been agreed, then any additional resources required should be allocated. All resources should have access to the restoration plan so that they can see what has been done so far and what actions they and other resources should now be working on.

The major incident manager should ensure that any necessary travel arrangements are authorized for support staff.

Once the major incident procedure is underway, the plans have been agreed, and resources allocated, then the standard incident life cycle of investigation, diagnosis, resolution, recovery, and closure should be followed. The difference with a major incident is that the incident is owned and closely coordinated by the major incident manager, and the restoration and coordination plans are regularly reviewed and maintained throughout the incident life cycle.

The restoration team should hold a review meeting at regular intervals to discuss progress, update the restoration plan, and issue a restoration team progress report. The major incident manager may attend some, but not necessarily all, of these review meetings. The interval between review meetings should vary depending on the amount of activity occurring. For example, during an intense phase of investigation, the meetings should be more frequent than during a later period when the staff is simply waiting for confirmation that the resolution actions have been successful.

Regular management team meetings should also be held to keep all parties informed, discuss progress, and decide when management escalation is required. The major incident manager should facilitate

all management team progress meetings. Again, the interval between review meetings should vary according to the level of activity.

It may become necessary to perform management escalation actions during the major incident. As the incident becomes more critical and as management escalation takes place within partners and customers, it is necessary to escalate awareness of the incident up the management chain. Situations, such as a senior manager or director first learning about the incident from their counterpart at a partner or customer organization, are to be avoided.

The communication plan should be regularly updated as management escalation takes place or restoration team members change. The major incident manager is responsible for either agreeing or obtaining agreement for all communication update statements prior to their release.

As the major incident progresses, the major incident manager should ensure that any additional resource requirements are located, allocated, and coordinated.

Some major incidents may necessitate the invocation of an organization's business continuity and service continuity plans. Consideration must be given as to how the major incident procedure and the service continuity plans work together and where responsibilities lie. During a major incident, the major incident manager should decide when and if the service continuity plan should be invoked in order to recover a service or to preserve evidence in the case of malicious intent. Normally, the major incident manager should remain responsible for coordinating the recovery operations, communications, and activities such as gathering evidence and implementing temporary workarounds.

Depending on their size and needs, organizations may make different levels of preparation for handling major incidents. Staffing issues, such as whether there are full-time major incident managers and an "on the bench" core restoration team, have already been mentioned, but some organizations may make additional preparations with dedicated "war rooms," pre-equipped with personal computers, communications equipment, and whiteboards. Many organizations realize that the public perception of how well they handled a major incident is directly related to how well they handle the media and so invest in training to ensure that spokespersons and senior managers can "deal with the media." Obviously, how reasonable it is to invest in preparations for handling major incidents depends on the size of the organization and the conditions under which it operates, but each organization should carefully consider the potential impacts that could occur and what they would mean for the organization, its staff, and its customers, while also realizing that sooner or later a major incident will occur. It is not a case of "if," but rather a case of "when."

The major incident procedure should continue alongside the normal incident management process until the major incident is closed. During the final phases of the incident, when resolution actions have been taken, it may be possible to release a portion of the restoration team from assignment, on the understanding that they will be recalled if the resolution proves to be unsuccessful.

At closure of major incidents, problem management should be notified, as it is their responsibility to carry out post major incident reviews.

Technology note: The staff responsible for responding to major incidents needs to be equipped accordingly. Communication is often difficult, since staff are working at remote sites or spending time in transit. Mobile communications technologies are required to ensure that the staff is in touch and can receive vital, often rapidly changing, information.

To aid communications plans, distribution lists can be set up for e-mails and text messages. Facilities such as digital display boards, closed circuit TV, and in-building radio broadcast systems can be used to provide all office-based staff with information about major incidents.

Support staff may need to view and update the incident record while working remotely. Service desk tools should be deployed that allow incidents to be viewed and updated remotely via browser or e-mail. Steps should be taken to ensure the security of these systems.

The major incident manager and other staff working on a major incident need to have access to both the communications plan and the restoration plan from wherever they are located. A portal solution allowing remote browser access to authorized staff can be used to provide this information.

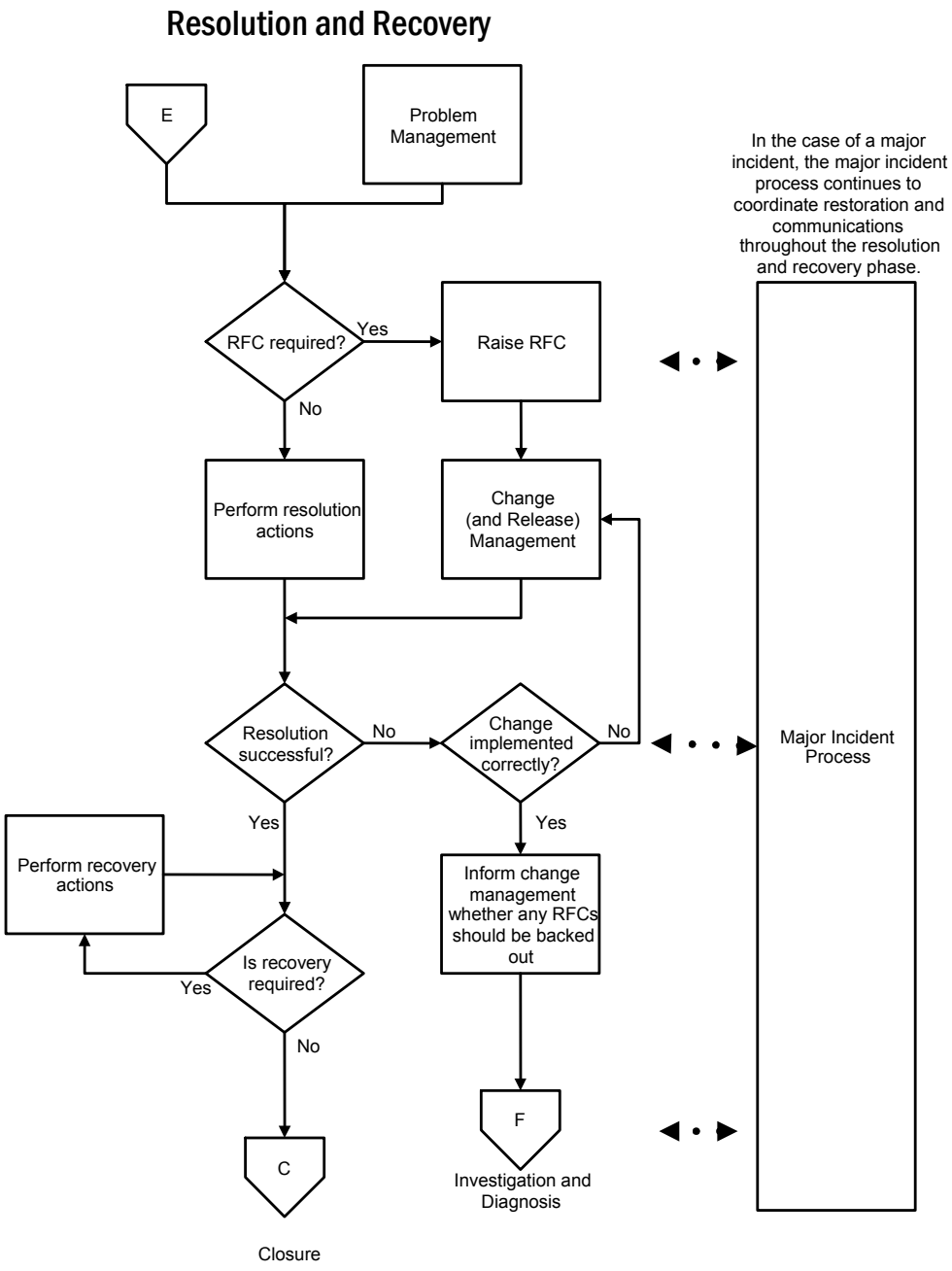


Figure 21

Resolution and recovery process flow chart

The resolution and recovery process is responsible for ensuring that any identified workarounds or solutions are properly implemented in accordance with change and release management processes and that any additional recovery actions are then taken.

Resolution actions are the actions that need to be taken to resolve the immediate cause of an incident, such as replacing a failed hard disk or installing a hot fix to a database application to prevent table corruption.

Recovery actions are the actions that *still* need to be taken once the resolution actions have been completed and all components are back in normal working order. In the case of a failed hard disk, the resolution action is to replace the disk, which resolves the cause of the incident. However, a recovery action of restoring the disk contents is still required. In the case of the database application, applying the hot fix is the resolution action that resolves the incident; however, recovery actions are still required to roll back the data to a consistent point and to restart the database service. Depending on the incident, the resolution and recovery actions may need to be carried out by different teams.

Incidents may arrive in the resolution and recovery phase because of a workaround or solution being identified during the investigation and diagnosis phase or because of problem management finding a workaround or resolution to a problem that had one or more outstanding incidents associated with it. In the latter scenario, problem management is responsible for identifying and testing the problem workaround or resolution and then informing incident management of the details. Incident management is then responsible for implementing the resolution, recovery, and closure of all incidents associated with the problem record.

Once the resolution actions have been identified, the incident management staff often needs to interface with the change management process to implement any changes required to resolve the incident. This ensures that the changes are properly tested and documented and that backout plans are considered. Incident management should monitor the progress of the resolution actions through the change and release management processes.

Some incidents might be resolved without the need to raise an RFC. These are often procedural issues where the incident was due to "human error" or incorrectly followed procedures.

Once the resolution actions have been carried out, incident management is responsible for confirming that the resolution actions have been successful. If they have not been successful, the first thing that should be checked is to verify that all changes were implemented correctly. If the changes are found to have been implemented incorrectly, then the change management process should be used to back out the failed changes and plan a new implementation.

If any RFCs were implemented correctly but the incident has not been resolved, then the incident should be passed back to the investigation and diagnosis process. Incident management staff should decide whether the implemented changes need to be backed out using the change management process. In many instances, the implemented changes may not require backing out, as they are beneficial in their own right. For example, the rollout of the latest service pack could be carried out in order to try and resolve a specific

incident. If the action did not resolve the specific intended incident, it would still be reasonable to retain the latest service pack as this should resolve or prevent other incidents.

Once resolution actions have been successfully implemented, then any recovery actions should be carried out. These recovery actions frequently involve making the service available to users again, following the resolution of the incident that caused the incident. Recovery actions should be carried out until the service is in a state where normal service has been restored.

If the incident in question was treated as a major incident, then the major incident procedure should continue alongside the resolution and recovery process, providing an enhanced level of coordination, communication, and planning.

Once the resolution and recovery actions have been carried out, the incident record should be placed in a status of "resolved" and assigned to the service desk analysts responsible for confirming resolution with the initiator.

Technology note: Integrated service management tool sets, which allow integration of the configuration management database (CMDB) with service requests, incident records, problem records, known errors, service level targets, and RFCs, give organizations an important tool for supporting their investment in service management processes.

The level of integration provided by these tools allows the full benefits of service management to be realized. Although the MOF SMFs can be implemented separately, the benefits can be more fully realized through a wider implementation, bringing an integrated set of processes that support and strengthen each other.

Tools can allow incident management staff to raise RFCs that are linked or associated with incident records, allowing progress to be tracked through a single interface rather than having to log on to multiple applications and re-input basic information.

Closure

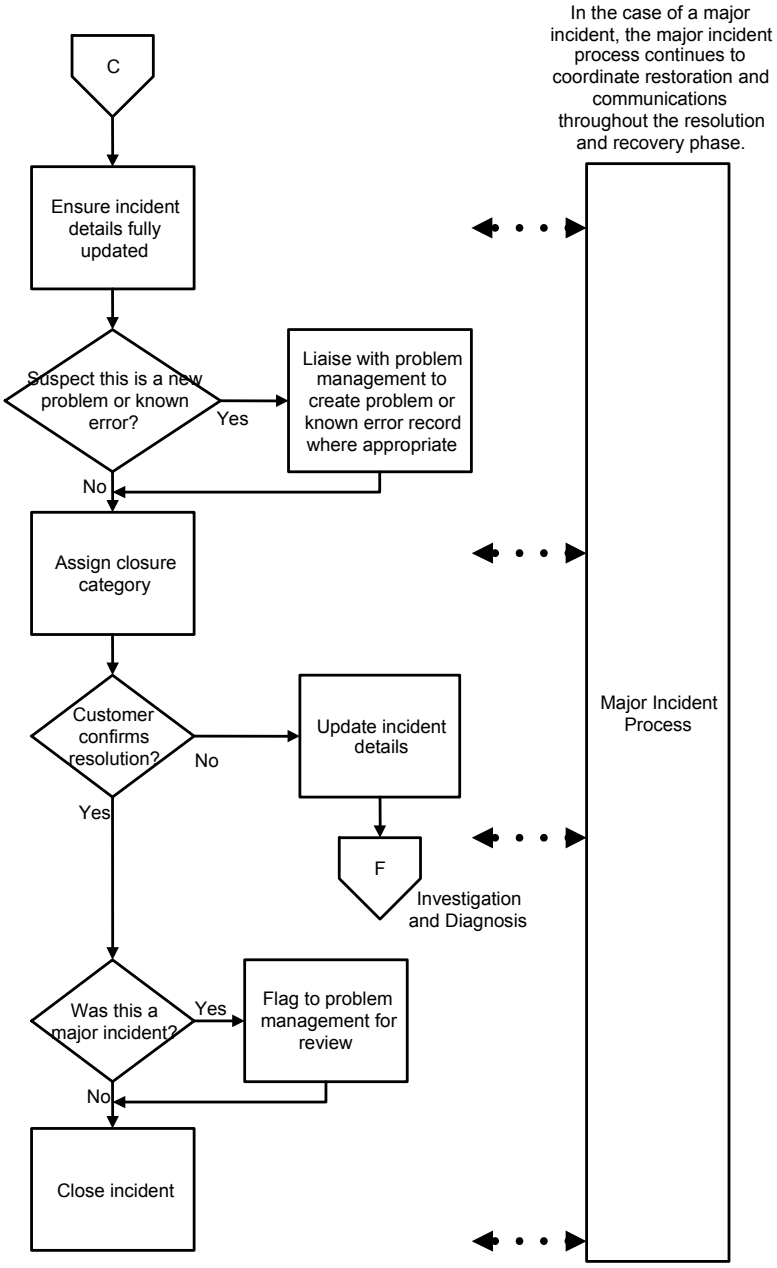


Figure 22

Closure process flow chart

The closure phase of the incident management process is responsible for confirming with the initiator that the incident had been resolved, ensuring that details of the incident and the resolution have been recorded, categorizing the closure and then closing the incident record.

It is the service desk analyst's responsibility to close incidents. Once incidents have been resolved, the resolver groups should update the incident record status to "resolved," and then pass the records back to the service desk team.

The service desk analysts should ensure that the incident record has been fully populated with details of the actions taken throughout the life cycle of the incident. Where information is felt to be lacking or unclear, the service desk analyst should contact the relevant resolver group in order to obtain an update. Where applicable, the service desk analyst should check that charging details, such as the cost center and the amount of time spent on the incident, have been recorded on the incident record.

If the incident is thought likely to happen again or has already had multiple occurrences but there is no corresponding problem or known error record, then the service desk analyst should liaise with problem management to request that a record be created. Problem management should consider the request and either create a new record or explain why one is not currently required.

The service desk analyst should also ensure that the incident has been allocated a closure category. The closure category should reflect the immediate cause of the incident. The actual closure categories in use should be agreed with problem management and be created at a sufficient level of detail so that useful metrics and reports can be obtained. For example, if there is a closure category of "faulty configuration," and it is reported that 50 incidents within the last month have been caused by this, problem management does not have much to go on. However, if this closure category had been broken down a bit further, to show "faulty PC configuration," "faulty server configuration,", "faulty account configuration," "faulty network configuration," and so on, the information would be much more useful. Problem management might now be able to see that 45 of the 50 incidents were caused by faulty account configurations, giving them a much better start toward identifying the root cause.

Technology note: Service desk tools can be configured to supply a list of potential closure categories to staff updating incident records. The closure category field should be made mandatory to ensure that a closure category has been selected prior to closure.

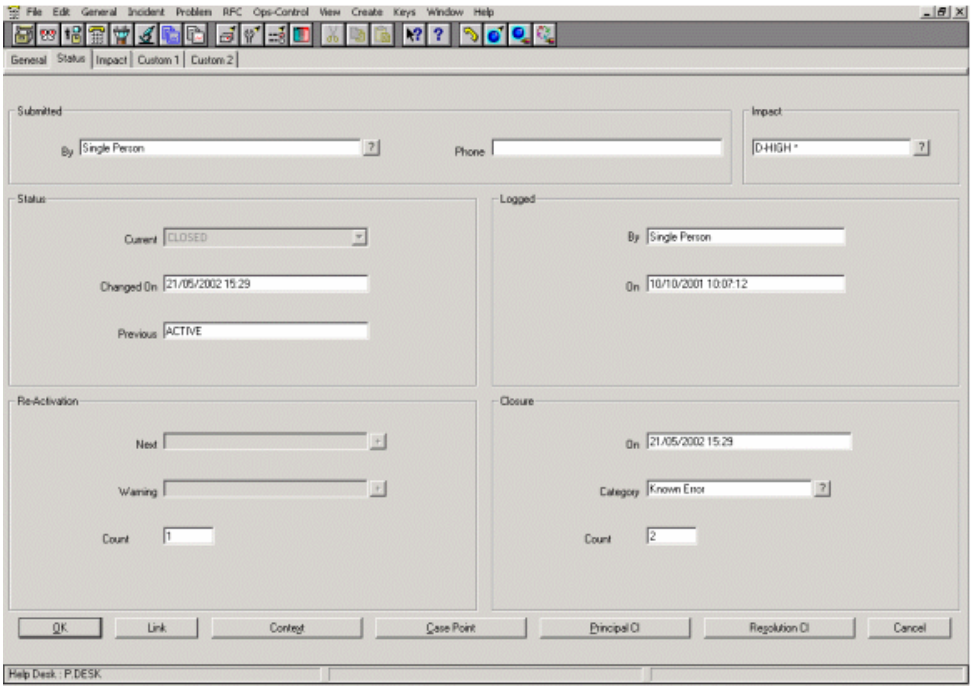


Figure 23

Example of an incident record closed with a closure category of "known error"

It is vital that the list of closure categories be comprehensive and well maintained. If staff cannot find a closure category that matches the incident they are dealing with, they can all too quickly fall into the habit of selecting a random category just so that the incident can be closed. For this reason, the closure category field should not be supplied with a default value. A closure category of "no relevant category" can be set up to give staff an option when they cannot see a closure category that matches their incident. To avoid this category being treated as a catchall, all instances of its use should be rigorously followed up. A regular report can be automatically produced for the incident manager and problem management staff showing instances when "no relevant category" has been selected. Each instance should be investigated and appropriate action taken either to set up a new category or to inform the staff concerned of the closure category that should have been used.

The situation should be monitored to ensure that closure categories are being used properly.

Finally, the service desk function should act as an "independent" check to verify that the initiator is happy with the support received and agrees that the incident has now been resolved. This check prevents situations where initiators call to check progress on an incident that is still impacting them, only to be told the incident has been closed because support staff thought they had resolved it. These situations are not only embarrassing for the support organization, but also have a very bad effect on customer satisfaction.

The service desk analysts should monitor for incidents being placed within a resolved state and then contact the initiator to confirm the resolution. Depending on the number of incidents being dealt with, this check may be done either by phone or by e-mail. Often a combination can be used—with phone calls being used for high-priority incidents and critical or sensitive services, while e-mails are used for standard, day-to-day incidents and users who are difficult to contact by telephone.

Technology note: Once resolution and recovery actions have been completed, incident records should be placed into a status of "resolved."

Tools may be configured to automatically send e-mail (using a standard e-mail template) to incident initiators once their incident has been placed in a status of resolved. The e-mails should contain the incident reference number, details of the date and time logged, and a brief description of the incident symptoms as first recorded. The e-mail should explain that the incident is now considered resolved, but if this is not the case or if the initiator is unhappy with the resolution, could he or she please either reply to the e-mail or contact the service desk via telephone. The e-mail should state that if nothing is heard within a certain period (perhaps several weeks to cover instances when an initiator is away on vacation, for example), then the incident will be closed. The tool should automatically close the incident if nothing is heard by the end of the specified period.

Tools should automatically notify problem management of any incidents that have been marked as major incidents. This can easily be done via e-mail.

In instances where the initiator reports that the incident has not been resolved satisfactorily, the service desk analyst should update the incident record, including the priority if this is now different, and then either re-assign the incident to a resolver group or escalate the incident. There will be instance when the support staff will have done all that they are able to do given their current resources, but the initiator is still not happy with the situation. Performance problems are a typical example of this—where the initiator is unhappy with the response times being obtained, but any tuning measures carried out by support staff have made little or no difference because the bottleneck is the current network infrastructure and significant investment would be needed to improve it. In this case there is little to be gained from passing the incident back to the resolver group, and the service desk analyst should instead escalate the incident either to the service manager responsible, if there is one, or otherwise directly to problem management. Obviously this is a case where having agreed service targets for response times can help, as long as their achievability has actually been properly established prior to their being agreed upon.

In the case of major incidents, the major incident procedure should run alongside the closure process, ensuring enhanced levels of coordination, communication, and planning. Problem management should be notified when major incidents are closed, as it is their responsibility to conduct reviews to establish both the root causes and whether the incident could have been handled differently.

Problem management should seek to identify how to prevent this incident or similar incidents from recurring.

Roles and Responsibilities

Principal roles and their associated responsibilities for incident management have been defined according to industry best practices. Organizations might need to combine some roles, depending on organizational size, organizational structure, and the underlying service level agreements existing between the IT department and the business it serves.

It is important to remember that these are roles rather than job descriptions.

The roles required within the incident management process are:

- Incident manager
- Service desk analyst
- Major incident manager
- Specialist support

Incident Manager

The role of incident manager includes responsibility for the end-to-end ownership of the incident management process in order to ensure that all incidents are progressed consistently according to their priority status. Otherwise, individual resolver groups might work to their own priorities, leading to a costly and ineffective support process. Where end-to-end visibility and responsibility does not exist, organizations can end up with each support tier working to their own set of priority codes and target resolution times, with no guiding structure as to how incidents will be handled, updated, or escalated. Frequently, this leads to disputes between the teams, resulting in low customer satisfaction and increased cost and staff frustration.

In many organizations, the incident manager and service desk manager roles are performed by the same person. Often the role is assigned to positions that are responsible for managing either first-line, or first- and second-line support teams.

In order for incidents to be escalated when required, an incident manager should always be available during service hours. During critical incidents, the incident manager is responsible for deciding whether to invoke the major incident procedure.

The incident manager should also be responsible for monitoring the performance of the incident management process and seeking to continuously improve the process.

Table 7. Incident Manager Responsibilities

Role	Main Responsibilities
Incident Manager	Driving the efficiency and effectiveness of the incident management process Producing management information May manage the work of first- and second-line support staff Developing and maintaining the incident management system The identification of major incidents

Service Desk Analysts

The service desk analysts are responsible for many tasks within the incident management process. During the initial phases of the incident life cycle, they are responsible for ensuring that the incident is properly recorded, classified, and given initial support. During initial support, they are responsible for resolving as many incidents as possible, within the timescales allowed. Their actions at this stage have a very direct impact on customer satisfaction and determine how the incidents will be dealt with by the rest of the support chain.

The service desk analysts are responsible for assigning incidents that have not been resolved during the initial support process to resolver groups. However, their responsibility does not end at this point and they should retain ownership of the incident, remaining responsible for ensuring that the incident continues to be progressed and escalated in accordance with service level targets.

The analysts are responsible for providing progress updates to customers throughout the life of an incident.

Once incidents have been resolved, the analysts should confirm that the initiator is satisfied with the incident resolution, prior to closing the incident record.

Table 8. Service Desk Analyst Responsibilities

Role	Main Responsibilities
Service Desk Analyst	Incident recording Initial support and classification Routing requests to resolver groups when incidents are not resolved during initial support Monitoring the status and progress toward resolution of all open incidents Keeping affected users informed about progress Escalating the process if necessary Resolution and recovery of incidents not assigned to resolver groups Resolution confirmation and closure of incidents

The role of the service desk analyst is described further within the Service Desk SMF.

Major Incident Manager

The role of major incident manager is critical during periods when major incidents are occurring. When a major incident occurs, a major incident manager should be immediately assigned to the incident. From then until the incident is closed, the major incident manager is responsible for ensuring a coordinated cross-company reaction, the assignment of resources to the restoration team, management escalation, and the management of communications with all parties involved.

The major incident manager is responsible for ensuring that both a restoration plan and a communications plan are produced and maintained throughout the incident.

The role needs to be assumed by a senior manager with an in-depth knowledge of the organization and the political authority to make things happen. The role involves much interaction with customers and can often be stressful. It requires both the ability to work under pressure and the ability to communicate with internal IT and organizational staff at all levels and with external partners and customers.

The number of major incident managers required and whether their commitment should be full-time or part-time depends on the size and nature of the organization. A starting point should be to look at historical major incidents: the length of time they took to resolve, how frequently they occurred, their location, the number that occurred simultaneously, and whether resolution work continued around the clock.

Table 7 - Major Incident Manager Responsibilities

Role	Main Responsibilities
Major Incident Manager	Coordination and management during major incidents Production and maintenance of the major incident communication plan Facilitating the production and maintenance of the major incident restoration plan Facilitation of management team reviews Production of major incident progress updates Participation in major incident reviews

Support Technicians

The Microsoft Operations Framework divides the IT environment into a set of layers: Each layer depends upon a firm foundation in order to function effectively. For the IT department, these layers are:

- Service
- Application
- Middleware
- Operating system
- Hardware
- Local area network
- Facilities
- Egress

Each layer has a support technician role that is responsible for participating in a resolver group whenever that layer is experiencing an incident. They are also actively involved in identifying problems and known errors.

Many individual teams within the IT organization may become involved with resolving different types of problems and known errors. These "resolver groups" should each have a queue within the problem management tool, to which problems requiring their specialist knowledge are assigned. Not all resolver groups will be purely "technical" groups, but they can all be seen as performing some form of specialist support function. Examples of "non-technical" resolver groups might be procedural experts or teams that are responsible for training, processing RFCs, or dealing with requests to change agreed service levels. The staff members within these groups are performing specialist support roles. Not all specialist support staff is necessarily internal, as external vendors may form a number of the resolver groups.

Staff members performing problem support roles are unlikely to have an in-depth knowledge of all areas within the IT infrastructure,

especially as new technologies continue to emerge. Staff members within specialist support roles are responsible for providing specialist knowledge in order to assist problem management in investigating and resolving problems.

When resolution actions are identified, specialist support staff should implement those actions under the control of the change and release management processes. Once resolution has been achieved, they should update the problem record and pass it back to problem support staff for closure.

Specialist support teams should also seek to identify underlying problems and notify problem management of the problems.

Table 8 - Support Technicians' Responsibilities

Role	Main Responsibilities
Application Support Technician	Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents Carrying out actions in order to correct known error
Middleware Support Technician	Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents Carrying out actions in order to correct known error

Storage Support Technician	Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents arrying out actions in order to correct known error Executes end-user data restoration requests.
Print Support Technician	Ensures that sufficient hardware spares are on-hand to meet service level requirements Creates printer standards to minimize spare parts requirements Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents arrying out actions in order to correct known error
Database Support Technician	Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents Carrying out actions in order to correct known error Executes end-user data restoration requests.

Directory Support Technician	Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents arrying out actions in order to correct known error Executes end-user data restoration requests.
Operating Support Technician	Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents arrying out actions in order to correct known error
Hardware Support Technician	Ensures that sufficient hardware spares are on-hand to meet service level requirements Manages the company's supply of spare parts Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents Carrying out actions in order to correct known error

Network Support Technician	Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents arrying out actions in order to correct known error
Facilities Support Technician	Handling service requests Monitoring incident details, including the configuration items affected Incident and problem investigation and diagnosis (including resolution where possible) Detection of possible problems and the notification of problem management The resolution and recovery of assigned incidents Acting as a restoration team member if required during major incidents arrying out actions in order to correct known error

Relationship to Other Processes

Service Desk

The service desk acts as the initial gateway to many of the IT processes, including the incident management process. With many of the other IT processes, the service desk simply passes requests to them; however, with the incident management process, the connection between the service desk and the incident management process is much closer.

The service desk acts as the interface between the business and IT, and in this case between the business and the incident management process.

The service desk manager is responsible for the day-to-day coordination of the incident management process. The service desk performs the recording, classification, and initial support phases of incident management. Then, when incidents are assigned to resolver groups, the service desk retains responsibility for ownership by monitoring and tracking all incidents.

If the service desk functions successfully, many service requests and incidents may be handled and resolved without ever going outside of the service desk function.

The service desk is a key component with regard to customer satisfaction with the incident management process.

Problem Management

Incident and problem management are closely related but have very different focuses within the support organization.

While incident management is focused on restoring normal service as quickly as possible, problem management is focused on the identification and resolution of underlying problems and their root causes.

Incident management provides problem management with much of the information that it requires in order to identify the existence of underlying problems. Problem management analyzes incident statistics to identify the occurrence of multiple incidents or increasing trends for specific types of incidents.

In return, problem management seeks to resolve the underlying causes of these incidents, to prevent their recurrence.

Problem management also maintains information about existing problems and known errors, including known workarounds and solutions. Incident management makes use of this information to provide a quicker resolution of incidents.

Problem management is also responsible for reviewing major incidents once they have been resolved. During these reviews, the aim is to identify underlying causes in order to prevent any recurrence, to identify triggers that can be used to give early warning of any recurrence, and to identify how well the incident was handled so as to improve future reactions.

Change Management

When incident management identifies resolution actions that require changes to configuration items, these changes are deployed under the control of the change management process.

The control provided by the change management process reduces the amount of changes that have to be backed out and ensures that backout plans have been documented in case they are required. The coordinated testing of changes reduces the chance of subsequent incidents caused by the deployment of changes.

Change management also provides incident management with useful information on recent changes, the configuration items affected, and the reason for the changes.

Configuration Management

Configuration management provides vital information that is used throughout the incident management process. The configuration management database (CMDB) contains information that can be used to:

- Provide and check caller details.
- Provide information on configuration items (CIs).
- Assist with the classification of incidents by indicating services and SLAs impacted by the failure of particular CIs.
- Identify the relationship and dependencies between CIs.
- Identify identical or similar CIs for comparison purposes.
- Identify alternative routes and workarounds.
- Record changes to configuration items because of RFCs.

Release Management

Some resolution actions identified by incident management require changes that need to be incorporated and rolled out as releases. These releases are coordinated and managed by the release management process.

In a similar way to change management, release management provides the effective planning, testing, and coordination that ensure these releases will be rolled out with minimal incident.

Service Level Management

Service level management provides incident management with the vital information needed to correctly classify new incidents. Information on business criticality and service level targets allows the incident management process to prioritize incidents, targeting support resources where most needed.

Incident management provides service level management with information on service target breaches.

Capacity Management

Capacity management assists incident management with ensuring that there is sufficient capacity for service desk tools, incident diagnosis tools, and self- service facilities.

Additionally, the planning provided by capacity management is aimed at preventing incidents occurring due to lack of resource (for example, CPU power or network bandwidth).

Availability Management

Availability management is interested in the impact on service availability caused by incidents. Incident management provides information on the number and duration of service breaks, including their cause and the actions taken to resolve them.

Availability management aims to reduce both the impact and likelihood of future service breaks by implementing prevention and mitigation measures.

Service Continuity Management

Service continuity management aims to prevent or reduce the impact of future incidents by implementing continuity measures and by making continuity plans.

During major incidents, the plans produced by service continuity management are used to recover failed services.

Security Administration

Security administration interfaces with the incident management process to ensure that procedures exist for handling security incidents. These procedures aim to ensure a fast response to any security-related incidents, while also ensuring that evidence is preserved in cases where malicious intent is suspected.

Security administration provides advice on the type and characteristics of evidence required for legal and disciplinary proceedings.

Security administration also reviews all security incidents, with the aim of preventing recurrence.

Service Monitoring and Control

Service monitoring and control provides the event management and alerting tools that provide information about and allow early detection of incidents. The SMF sets thresholds that allow incident management to react to incidents before service targets are breached.

Service monitoring and control tools may be integrated with service desk tools to provide automatic generation of incident records based on alerts received.

Appendixes

Appendix A: Technologies Used

All of the screenshots used within this document with the exception of the knowledge base example, are taken from Fox IT's Redbox service management application, which integrates the functions of configuration management, change management, service desk, incident management, problem management and operations control to assist organizations in providing world-class support for their IT services.

The knowledge base screenshot is taken from Microsoft's TechNet support application.

Appendix B: Example Major Incident Restoration Plan Template

Incident Overview

This section contains the basic incident details.

Incident Reference ID	
Date/Time Incident Occurred	
Date/Time Incident Recorded	
Date/Time Major Incident Declared	
Contact Details of Initiator	
Incident Status	
Main Service Affected	
Incident Summary	

Description of Impact and Deadlines

This section includes a full description of the impact, services affected, service targets threatened/breached, and any associated deadlines.

Services, customers, and locations affected
Service targets involved
Description of impacts
Explanation of any deadlines/timescales involved

Roles and Contact Details

This section provides contact details for the staff assigned to work on the incident.

Personnel	Position	Location	Main Contact Number	Secondary Contact Numbers	E-mail
Major Incident Manager(s)					
Principal Business Contact(s)					

Restoration Team Leader(s)					
Restoration Team Personnel					
Additional Contacts					

Incident Investigation

Incident Description Statement

This section should include a statement of what the difficulty is currently perceived to be. This statement may evolve as more is learned about the incident.

Incident Detail

This section should detail information about the incident, including:

- Symptoms
- CIs affected/involved
- Service descriptions
- Any relevant diagrams showing network infrastructure/data flow

Incident Event Timeline

This section provides a timeline for events during the incident. It is often possible to build up a clearer picture of "what went wrong" by placing the events and error messages into chronological order.

Date/Time	Event

Possible Immediate Causes

This section should record ideas on the possible immediate causes of the incident. Often determining the immediate cause allows one to plan how to resolve an incident. For example, if the web site is down, has a denial of service attack occurred? These possible causes may not turn out to be the actual cause but help to provide lines of investigation so that probable causes can be ruled out. A brainstorming session might be conducted in order to populate the

table. Note that these are immediate causes of the failure rather than any underlying root causes. The ideas should be rated in order to prioritize those that are thought most likely to be correct.

Possible Cause	**Likelihood** (rated 1-10)	**How Can We Investigate to Rule This Out?**

Potential Resolution Actions

This section should record ideas about potential resolution actions. For example, if a server is experiencing network connection difficulties, is there a later driver for the network adaptor card that would fix the issue? These actions may not turn out to actually resolve the incident, but provide lines of investigation that can be pursued. A brainstorming session might be conducted in order to populate the table. The actions should be rated in order to prioritize those that are thought most likely to be successful.

Potential Resolution Action	**Likelihood** (rated 1-10)	**How Can We Test /Confirm This?**

Action Plan

This section should detail the actions that are planned based on the ideas captured within the previous section. The actions should be prioritized based upon the likelihood ratings.

Action	Personnel Allocated	Scheduled Date/Time for Action	Date/Time Action to Be Completed By	Status

Completed Actions

As planned actions are completed, they should be documented within this section and any follow-up actions that have been identified should be added to the outstanding action plan.

Action	Personnel Allocated	Date/Time Action Completed	Results

Next Review

This section should be used to document the date, time, and location of the next restoration team review, including any details such as teleconference joining instructions.

Appendix C: Example Major incident Communication Plan Matrix

Communication Plan Matrix					
Incident Number		**Major Incident Manager**		**Date Plan Created**	
Matrix Version Number		**Date/Time Next Management Meeting Due**		**Plan Last Updated**	
Statement Type	**Produced By**	**Statement (link to statement document)**	**Authorized By**	**Date/Time Issued**	**Next Due Date/ Time**
Recipient Name	**Organization**	**Contact Details**	**Update Type Required**	**Frequency**	**Method**

Contributors

Many of the practices that this document describes are based on years of IT implementation experience by Accenture, Avanade, Microsoft Consulting Services, Fox IT, Hewlett-Packard Company, Lucent Technologies/NetworkCare Professional Services, and Unisys Corporation.

Microsoft gratefully acknowledges the generous assistance of these organizations in providing material for this document.

Program Management Team

William Bagley, Microsoft Corporation

Neil Battell, Fox IT

Lead Writer

Neil Battell, Fox IT

Contributing Writer

Tony Brooks, Fox IT

Editor

Patricia Rytkonen, Volt Technical Services

Other Contributors

Steve McReynolds, Microsoft Corporation

Dave Phillips, Compuware